

Package ‘cplexAPI’

September 21, 2020

Type Package

Title R Interface to C API of IBM ILOG CPLEX

Version 1.4.0

Date 2020-09-17

Depends R (>= 2.6.0)

Imports methods

Description This is the R Interface to the C API of IBM ILOG CPLEX. It necessarily depends on IBM ILOG CPLEX (>= 12.1).

SystemRequirements IBM ILOG CPLEX (>= 12.1)

License GPL-3 | file LICENSE

LazyLoad yes

Collate generics.R cplexConst.R cplexErrorClass.R cplexPtrClass.R
cplex.R cplexAPI.R cplex_checkAPI.R cplex_longparamAPI.R zzz.R

NeedsCompilation yes

Repository CRAN

Date/Publication 2020-09-21 16:20:11 UTC

Author Mayo Roettger [cre],
Gabriel Gelius-Dietrich [aut],
C. Jonathan Fritzemeier [ctb]

Maintainer Mayo Roettger <mayo.roettger@hhu.de>

R topics documented:

cplexAPI-package	6
addColsCPLEX	7
addFpDestCPLEX	9
addIndConstrCPLEX	10
addMIPstartsCPLEX	11
addQConstrCPLEX	12
addRowsCPLEX	13
baroptCPLEX	14

baseWriteCPLEX	15
basicPresolveCPLEX	16
boundSaCPLEX	17
checkAddColsCPLEX	18
checkAddRowsCPLEX	19
checkChgCoefListCPLEX	20
checkCopyColTypeCPLEX	21
checkCopyLpCPLEX	22
checkCopyLpwNamesCPLEX	23
checkCopyQPsepCPLEX	25
checkCopyQuadCPLEX	26
checkValsCPLEX	27
chgBndsCPLEX	28
chgCoefCPLEX	29
chgCoefListCPLEX	30
chgColNameCPLEX	31
chgColsBndsCPLEX	32
chgColTypeCPLEX	33
chgMIPstartsCPLEX	34
chgNameCPLEX	35
chgObjCPLEX	36
chgProbNameCPLEX	37
chgProbTypeCPLEX	38
chgQPcoefCPLEX	39
chgRhsCPLEX	40
chgRngValCPLEX	41
chgRowNameCPLEX	42
chgSenseCPLEX	43
chgTerminateCPLEX	44
cleanupCoefCPLEX	44
cloneProbCPLEX	45
closeEnvCPLEX	46
closeFileCPLEX	47
closeProbCPLEX	48
cLpWriteCPLEX	49
completlpCPLEX	50
copyBaseCPLEX	51
copyColTypeCPLEX	52
copyLpCPLEX	53
copyLpwNamesCPLEX	54
copyObjNameCPLEX	55
copyOrderCPLEX	56
copyQPsepCPLEX	57
copyQuadCPLEX	58
copyStartCPLEX	59
cplexConstants	60
cplexError-class	82
cplexPtr-class	83

delColsCPLEX	84
delFpDestCPLEX	85
delIndConstrsCPLEX	86
delMIPstartsCPLEX	87
delNamesCPLEX	88
delProbCPLEX	89
delQConstrsCPLEX	90
delRowsCPLEX	91
delSetColsCPLEX	92
delSetRowsCPLEX	93
delTerminateCPLEX	94
disconnectChannelCPLEX	95
dualoptCPLEX	96
dualWriteCPLEX	97
feasOptCPLEX	98
fileputCPLEX	99
flushChannelCPLEX	100
flushStdChannelsCPLEX	101
freePresolveCPLEX	102
getBaseCPLEX	103
getBestObjValCPLEX	104
getChannelsCPLEX	105
getChgParmCPLEX	106
getCoefCPLEX	107
getColIndexCPLEX	108
getColInfeasCPLEX	109
getColNameCPLEX	110
getColsCPLEX	111
getColTypeCPLEX	112
getConflictCPLEX	113
getConflictExtCPLEX	114
getCutoffCPLEX	115
getDbIParmCPLEX	116
getDbIQualCPLEX	117
getDbSCntCPLEX	118
getDjCPLEX	119
getErrorStrCPLEX	120
getGradCPLEX	121
getIndConstrCPLEX	122
getInfoDbIParmCPLEX	123
getInfoIntParmCPLEX	124
getInfoLongParmCPLEX	125
getInfoStrParmCPLEX	126
getIntParmCPLEX	127
getIntQualCPLEX	128
getItCntCPLEX	129
getLogFileCPLEX	130
getLogFileNameCPLEX	131

getLongParmCPLEX	132
getLowBndsIdsCPLEX	133
getLowerBndsCPLEX	134
getMethodCPLEX	135
getMIPrelGapCPLEX	136
getMIPstartIndexCPLEX	137
getMIPstartNameCPLEX	138
getMIPstartsCPLEX	139
getNumColsCPLEX	140
getNumMIPstartsCPLEX	141
getNumNnzCPLEX	142
getNumQConstrsCPLEX	143
getNumQPnzCPLEX	144
getNumQuadCPLEX	145
getNumRowsCPLEX	146
getObjCPLEX	147
getObjDirCPLEX	148
getObjNameCPLEX	149
getObjOffsetCPLEX	150
getObjValCPLEX	151
getOrderCPLEX	152
getParmHierNameCPLEX	153
getParmNameCPLEX	154
getParmNumCPLEX	155
getParmTypeCPLEX	156
getParmValCPLEX	157
getPhase1CntCPLEX	157
getPiCPLEX	158
getPreStatCPLEX	159
getProbNameCPLEX	160
getProbTypeCPLEX	161
getProbVarCPLEX	162
getQConstrCPLEX	163
getQPcoefCPLEX	164
getQuadCPLEX	165
getRedLpCPLEX	166
getRhsCPLEX	167
getRngValCPLEX	168
getRowIndexCPLEX	169
getRowInfeasCPLEX	170
getRowNameCPLEX	171
getRowsCPLEX	172
getSenseCPLEX	173
getSiftItCntCPLEX	174
getSiftPhase1CntCPLEX	175
getSlackCPLEX	176
getStatCPLEX	177
getStatStrCPLEX	178

getStrParmCPLEX	179
getSubMethodCPLEX	180
getSubStatCPLEX	181
getTimeCPLEX	182
getUppBndsIdsCPLEX	183
getUpperBndsCPLEX	184
getVersionCPLEX	185
hybbaroptCPLEX	185
hybnetoptCPLEX	186
initProbCPLEX	187
lpoptCPLEX	188
mipoptCPLEX	189
newColsCPLEX	190
newRowsCPLEX	191
objSaCPLEX	192
openEnvCPLEX	193
openFileCPLEX	194
openProbCPLEX	195
ordWriteCPLEX	196
preslvWriteCPLEX	197
presolveCPLEX	198
primoptCPLEX	199
printTerminateCPLEX	200
qpoptCPLEX	200
readCopyBaseCPLEX	201
readCopyMIPstartsCPLEX	202
readCopyOrderCPLEX	203
readCopyParmCPLEX	204
readCopyProbCPLEX	205
readCopySolCPLEX	206
refineConflictCPLEX	207
refineConflictExtCPLEX	208
refineMIPstartConflictCPLEX	209
refineMIPstartConflictExtCPLEX	210
return_codeCPLEX	211
rhsSaCPLEX	212
setDbiParamCPLEX	213
setDefaultParmCPLEX	214
setIntParmCPLEX	215
setLogFileCPLEX	216
setLogFileNameCPLEX	217
setLongParmCPLEX	218
setObjDirCPLEX	219
setStrParmCPLEX	220
setTerminateCPLEX	221
siftoptCPLEX	222
solnInfoCPLEX	223
solutionCPLEX	224

solWriteCPLEX	225
status_codeCPLEX	226
tightenBndsCPLEX	227
tuneParmCPLEX	228
unscaleProbCPLEX	229
writeMIPstartsCPLEX	230
writeParmCPLEX	231
writeProbCPLEX	232

Index	233
--------------	------------

cplexAPI-package *R Interface to C API of IBM ILOG CPLEX*

Description

A low level interface to IBM ILOG CPLEX.

Details

The package cplexAPI provides access to the callable library of IBM ILOG CPLEX from within R.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

Examples

```
# load package
library(cplexAPI)

# Open a CPLEX environment
env <- openEnvCPLEX()

# Create a problem object
prob <- initProbCPLEX(env)

# Assign a name to the problem object
chgProbNameCPLEX(env, prob, "sample")

# Prepare data structures for the problem object
# Number of columns and rows
nc <- 3
```

```

nr <- 3

# Objective function
obj <- c(5, 4, 3)

# Right hand side
rhs <- c(5, 11, 8)

# Sense of the right hand side
sense <- rep("L", 3)

# Variable lower bounds
lb <- rep(0, 3)

# Variable upper bounds
ub <- rep(CPX_INFBOUND, 3)

# Column and row names
cn <- c("x1", "x2", "x3")
rn <- c("q1", "q2", "q3")

# The constraint matrix is passed in column major order format
# Be careful here: all indices start with 0! Begin indices of rows
beg <- c(0, 3, 6)

# Number of non-zero elements per row
cnt <- rep(3, 3)

# Column indices
ind <- c(0, 1, 2, 0, 1, 2, 0, 1, 2)

# Non-zero elements
val <- c(2, 4, 3, 3, 1, 4, 1, 2, 2)

# Load problem data
copyLpwNamesCPLEX(env, prob, nc, nr, CPX_MAX, obj, rhs, sense,
                  beg, cnt, ind, val, lb, ub, NULL, cn, rn)

# Solve the problem using the simplex algorithm
lpoptCPLEX(env, prob)

# Retrieve solution after optimization
solutionCPLEX(env, prob)

# Free memory, allacated to the problem object
delProbCPLEX(env, prob)
closeEnvCPLEX(env)

```

Description

Low level interface function to the IBM ILOG CPLEX function CPXaddcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
addColsCPLEX(env, lp, ncols, nnz, objf, matbeg, matind, matval,  
             lb = NULL, ub = NULL, cnames = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
ncols	Number of columns.
nnz	Number of nonzero constraint coefficients.
objf	Objective function coefficients.
matbeg	Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind	Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval	Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
lb	Lower bounds of the new variables.
ub	Upper bounds of the new variables.
cnames	Names of the new variables.

Details

Interface to the C function addCols which calls the CPLEX function CPXaddcols.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[checkAddColsCPLEX](#), [addRowsCPLEX](#)

addFpDestCPLEX *Add a File to the List of Message Destinations for a Channel*

Description

Low level interface function to the IBM ILOG CPLEX function CPXaddfpdest. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXaddfpdest has been removed.

Usage

```
addFpDestCPLEX(env, newch, cpfile)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
newch	Pointer to an IBM ILOG CPLEX channel as returned by addChannelCPLEX .
cpfile	Pointer to an IBM ILOG CPLEX file as returned by openFileCPLEX .

Details

Interface to the C function addFpDest which calls the CPLEX function CPXaddfpdest.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[delFpDestCPLEX](#)

addIndConstrCPLEX *Adds an Indicator Constraint to the Specified CPLEX Problem Object*

Description

Low level interface function to the IBM ILOG CPLEX function CPXaddindconstr. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
addIndConstrCPLEX(env, lp, indvar, complemented,
                  nzcnt, rhs, sense, linind, linval, indname = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
indvar	The binary variable that acts as the indicator for this constraint.
complemented	A Boolean value that specifies whether the indicator variable is complemented.
nzcnt	An integer that specifies the number of nonzero coefficients in the linear portion of the indicator constraint.
rhs	The righthand side value for the linear portion of the indicator constraint.
sense	The sense of the linear portion of the indicator constraint.
linind	A vector that with <code>linval</code> defines the linear portion of the indicator constraint.
linval	A vector that with <code>linind</code> defines the linear portion of the indicator constraint.
indname	The name of the constraint to be added (optional).

Details

Interface to the C function `addIndConstr` which calls the CPLEX function `CPXaddindconstr`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

addMIPstartsCPLEX *Add Multiple MIP Starts to a CPLEX Problem Object*

Description

Low level interface function to the IBM ILOG CPLEX function CPXaddmipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
addMIPstartsCPLEX(env, lp, mcnt, nzcnt, beg, varindices,  
                  values, effortlevel, mipstartname = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
mcnt	Number of MIP starts to be added.
nzcnt	Number of variable values to be added.
beg	Array of length mcnt used with varindices and values. Consult the IBM ILOG CPLEX documentation for more detailed information.
varindices	Array of length nzcnt containing the numeric indices of the columns corresponding to the variables which are assigned starting values. Consult the IBM ILOG CPLEX documentation for more detailed information.
values	Array of length nzcnt containing the values to use for the MIP starts. Consult the IBM ILOG CPLEX documentation for more detailed information.
effortlevel	Array of length mcnt. Consult the IBM ILOG CPLEX documentation for more detailed information.
mipstartname	Names of the MIP starts.

Details

Interface to the C function addMIPstarts which calls the CPLEX function CPXaddmipstarts.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

addQConstrCPLEX	<i>Add Quadratic Constraint to a Specified CPLEX Problem Object</i>
-----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXaddqconstr. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
addQConstrCPLEX(env, lp, lzn, qzn, rhs, sense,
                lind = NULL, lval = NULL,
                qrow, qcol, qval, qname = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
lzn	Number of nonzero constraint coefficients in the linear part of the constraint.
qzn	Number of nonzero constraint coefficients in the quadratic part of the constraint.
rhs	Righthand side term.
sense	The sense of the constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
lind	Linear part of the quadratic constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
lval	Linear part of the constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
qrow	Quadratic part of the quadratic constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
qcol	Quadratic part of the quadratic constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
qval	Quadratic part of the quadratic constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
qname	Name of the constraint to be added.

Details

Interface to the C function addQConstr which calls the CPLEX function CPXaddqconstr.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

 addRowsCPLEX

Add Constraints to a Specified CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXaddrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
addRowsCPLEX(env, lp, ncols, nrows, nnz, matbeg, matind, matval,
             rhs = NULL, sense = NULL,
             cnames = NULL, rnames = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
ncols	Number of new columns in the constraints being added to the constraint matrix.
nrows	Number of rows.
nnz	Number of nonzero constraint coefficients.
matbeg	An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind	An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval	An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
rhs	Righthand side term for each constraint to be added.
sense	Sense of each constraint to be added.
cnames	Names of the new columns.
rnames	Names of the new rows.

Details

Interface to the C function `addCols` which calls the CPLEX function `CPXaddcols`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[checkAddRowsCPLEX](#), [addColsCPLEX](#), [copyLpCPLEX](#), [chgRngValCPLEX](#)

baroptCPLEX

Solve LP, QP or QCP Problem by Means of the Barrier Algorithm

Description

Low level interface function to the IBM ILOG CPLEX function `CPXbaropt`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
baroptCPLEX(env, lp)
```

Arguments

`env` An object of class "`cplexPtr`" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.

`lp` An object of class "`cplexPtr`" as returned by [initProbCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `baropt` which calls the CPLEX function `CPXbaropt`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[solnInfoCPLEX](#), [getStatCPLEX](#), [solutionCPLEX](#)

baseWriteCPLEX	<i>Write the Most Current Basis Associated With a CPLEX Problem Object to a File</i>
----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function `CPXmbasewrite`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
baseWriteCPLEX(env, lp, fname)
```

Arguments

env	An object of class " <code>cplexPtr</code> " as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class " <code>cplexPtr</code> " as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
fname	A filename.

Details

Interface to the C function `baseWrite` which calls the CPLEX function `CPXmbasewrite`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

basicPresolveCPLEX *Perform Bound Strengthening and Detect Redundant Rows*

Description

Low level interface function to the IBM ILOG CPLEX function CPXbasicpresolve. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
basicPresolveCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `basicPresolve` which calls the CPLEX function `CPXbasicpresolve`.

Value

If successful, a list will be returned:

redlb	strengthened lower bounds
redub	strengthened upper bounds
rstat	status of the row

Otherwise an object of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

boundSaCplex	<i>Access Upper and Lower Sensitivity Ranges for Lower and Upper Variable Bounds</i>
--------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXboundsa. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
boundSaCplex(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCplex</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCplex</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	Beginning of the range of ranges to be returned.
end	End of the range of ranges to be returned.

Details

Interface to the C function boundSa which calls the CPLEX function CPXboundsa.

Value

If successful, a list will be returned:

lblower	lower bound lower range values
lbupper	lower bound upper range values
ublower	upper bound lower range values
ubupper	upper bound upper range values

Otherwise an object of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

checkAddColsCPLEX *Validate Arguments of the Corresponding addColsCPLEX Routine*

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckaddcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
checkAddColsCPLEX(env, lp, ncols, nnz, objf, matbeg, matind, matval,
                  lb = NULL, ub = NULL, cnames = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
ncols	Number of columns.
nnz	Number of nonzero constraint coefficients.
objf	Objective function coefficients.
matbeg	Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind	Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval	Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
lb	Lower bounds of the new variables.
ub	Upper bounds of the new variables.
cnames	Names of the new variables.

Details

Interface to the C function checkAddCols which calls the CPLEX function CPXcheckaddcols.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[addColsCPLEX](#)

checkAddRowsCPLEX *Validate Arguments of the Corresponding addRowsCPLEX Routine*

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckaddrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
checkAddRowsCPLEX(env, lp, ncols, nrows, nnz, matbeg, matind, matval,
                  rhs = NULL, sense = NULL,
                  cnames = NULL, rnames = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
ncols	Number of new columns in the constraints being added to the constraint matrix.
nrows	Number of rows.
nnz	Number of nonzero constraint coefficients.
matbeg	An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind	An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval	An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
rhs	Righthand side term for each constraint to be added.
sense	Sense of each constraint to be added.
cnames	Names of the new columns.
rnames	Names of the new rows.

Details

Interface to the C function checkAddRows which calls the CPLEX function CPXcheckaddrows.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[addRowsCPLEX](#)

checkChgCoefListCPLEX *Validate Arguments of the Corresponding chgCoefListCPLEX Routine*

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckchgcoeflist. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
checkChgCoefListCPLEX(env, lp, nnz, ia, ja, ra)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
nnz	Number of nonzero constraint coefficients.
ia	Row indices of the nonzero elements.
ja	Column indices of the nonzero elements.
ra	Nonzero elements.

Details

Interface to the C function checkChgCoefList which calls the CPLEX function CPXcheckchgcoeflist.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[chgCoefListCPLEX](#)

checkCopyColTypeCPLEX *Validate Arguments of the Corresponding copyColTypeCPLEX Routine*

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckcopyctype. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
checkCopyColTypeCPLEX(env, lp, xtype)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
xctype	A vector containing the type of each column in the constraint matrix.

Details

Interface to the C function checkCopyColType which calls the CPLEX function CPXcheckcopyctype.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[copyColTypeCPLEX](#)

checkCopyLpCPLEX

Validate Arguments of the Corresponding copyLpCPLEX Routine

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckcopylp. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
checkCopyLpCPLEX(env, lp, nCols, nRows, lmdir, objf, rhs, sense,
                 matbeg, matcnt, matind, matval, lb, ub, rngval = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
nCols	Number of columns in the constraint matrix.
nRows	Number of rows in the constraint matrix.
lmdir	Single integer value that specifies whether the problem is a minimization or maximization problem.
objf	The objective function coefficients.
rhs	The righthand side values for each constraint in the constraint matrix.
sense	The sense of each constraint in the constraint matrix.
matbeg	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matcnt	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
lb	Containing the lower bound on each of the variables.
ub	Containing the lower bound on each of the variables.
rngval	Containing the range value of each ranged constraint.

Details

Interface to the C function checkCopyLp which calls the CPLEX function CPXcheckcopylp.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[copyLpCPLEX](#)

checkCopyLpwNamesCPLEX

Validate Arguments of the Corresponding copyLpwNamesCPLEX Routine

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckcopylpwnames. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
checkCopyLpwNamesCPLEX(env, lp, nCols, nRows, lpdire, objf, rhs, sense,
                        matbeg, matcnt, matind, matval, lb, ub,
                        rngval = NULL, cnames = NULL, rnames = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
nCols	Number of columns in the constraint matrix.
nRows	Number of rows in the constraint matrix.
lpdire	Single integer value that specifies whether the problem is a minimization or maximization problem.

objf	The objective function coefficients.
rhs	The righthand side values for each constraint in the constraint matrix.
sense	The sense of each constraint in the constraint matrix.
matbeg	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matcnt	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
lb	Containing the lower bound on each of the variables.
ub	Containing the lower bound on each of the variables.
rngval	Containing the range value of each ranged constraint.
cnames	Names of the matrix columns or, equivalently, the variable names.
rnames	Names of the matrix rows or, equivalently, the constraint names.

Details

Interface to the C function `checkCopyLpwNames` which calls the CPLEX function `CPXcheckcopylpwnames`.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[copyLpwNamesCPLEX](#)

checkCopyQPsepCPLEX *Validate Arguments of the Corresponding copyQPsepCPLEX Routine*

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckcopyqpsep. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
checkCopyQPsepCPLEX(env, lp, qsepvec)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
qsepvec	A vector containing the quadratic coefficients.

Details

Interface to the C function checkCopyQPsep which calls the CPLEX function CPXcheckcopyqpsep.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[copyQPsepCPLEX](#)

checkCopyQuadCPLEX	<i>Validate Arguments of the Corresponding checkCopyQuadCPLEX Routine</i>
--------------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckcopyquad. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
checkCopyQuadCPLEX(env, lp, qmatbeg, qmatcnt, qmatind, qmatval)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
qmatbeg	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
qmatcnt	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
qmatind	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
qmatval	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.

Details

Interface to the C function `checkCopyQuad` which calls the CPLEX function `CPXcheckcopyquad`.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[copyQuadCPLEX](#)

checkValsCPLEX	<i>Check an Array of Indices and a Corresponding Array of Values for Input Errors</i>
----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckvals. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
checkValsCPLEX(env, lp, nval, rind = NULL, cind = NULL, val = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
nval	Number of values to be examined.
rind	Row indices.
cind	Column indices.
val	The values itself.

Details

Interface to the C function `checkVals` which calls the CPLEX function `CPXcheckvals`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`chgBndsCPLEX`*Change the Lower or Upper Bounds on a Set of Variables of a Problem*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgbd. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgBndsCPLEX(env, lp, ncols, ind, lu, bd)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>ncols</code>	Number of bounds to be changed.
<code>ind</code>	Indices of bounds to be changed.
<code>lu</code>	A character vector, specifying whether an entry in <code>bd</code> is a upper or a lower bound on variable <code>ind[j]</code> .
<code>bd</code>	Values of the lower or upper bounds of the variables present in <code>ind</code> .

Details

Interface to the C function `chgBnds` which calls the CPLEX function `CPXchgbd`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[getLowerBndsCPLEX](#), [getUpperBndsCPLEX](#)

chgCoefCPLEX	<i>Change a Single Coefficient in the Constraint Matrix, Linear Objective Coefficients, Righthand Side, or Ranges of a CPLEX Problem Object</i>
--------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgcoef. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgCoefCPLEX(env, lp, i, j, val)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
i	An integer that specifies the numeric index of the row in which the coefficient is located. The linear objective row is referenced with $i = -1$.
j	An integer that specifies the numeric index of the column in which the coefficient is located. The RHS column is referenced with $j = -1$. The range value column is referenced with $j = -2$. If $j = -2$ is specified and row i is not a ranged row, an error status is returned.
val	The new value for the coefficient being changed.

Details

Interface to the C function `chgCoef` which calls the CPLEX function `CPXchgcoef`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[getNumRowsCPLEX](#), [getNumColsCPLEX](#), [chgObjCPLEX](#), [chgRhsCPLEX](#), [chgRngValCPLEX](#)

`chgCoefListCPLEX`*Change a List of Matrix Coefficients of a CPLEX Problem Object*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXchgcoeflist`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgCoefListCPLEX(env, lp, nnz, ia, ja, ra)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>nnz</code>	Number of nonzero constraint coefficients.
<code>ia</code>	Row indices of the nonzero elements.
<code>ja</code>	Column indices of the nonzero elements.
<code>ra</code>	Nonzero elements.

Details

Interface to the C function `chgcoeflist` which calls the CPLEX function `CPXchgcoeflist`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

chgColNameCPLEX *Change the Names of Variables in a CPLEX Problem Object*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgcolname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgColNameCPLEX(env, lp, nnames, ind, names)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
nnames	A vector that specifies the total number of variable names to be changed.
ind	A vector containing the numeric indices indices of the variables for which the names are to be changed.
names	A vector containing the strings of the new variable names for the columns specified in ind.

Details

Interface to the C function chgColName which calls the CPLEX function CPXchgcolname.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

chgColsBndsCPLEX *Change Lower and Upper Bounds on a Set of Variables of a Problem*

Description

Set lower and upper bounds on a set of variables in one step. If `lb[i] == ub[i]` the type of the bound is set to "B", otherwise `lb[i]` is set to "L" and `ub[i]` is set to "U".

Usage

```
chgColsBndsCPLEX(env, lp, j, lb, ub)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
j	An integer that specifies the numeric index of the column in which the coefficient is located.
lb	A vector containing the lower bounds.
ub	A vector containing the upper bounds.

Details

Interface to the C function `chgColsBnds` which calls the CPLEX function `CPXchgbds`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

`chgBndsCPLEX`, `tightenBndsCPLEX`

chgColTypeCPLEX *Change Types of a Set of Variables of a CPLEX Problem Object*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgctype. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgColTypeCPLEX(env, lp, ncols, ind, xtype)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
ncols	Number of bounds to be changed.
ind	Indices of bounds to be changed.
xctype	A vector containing characters that represent the new types for the columns specified in indices.

Details

Interface to the C function chgColType which calls the CPLEX function CPXchgctype.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

chgMIPstartsCPLEX *Modify or Extend Multiple MIP Starts*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchg mipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgMIPstartsCPLEX(env, lp, mcnt, mipstartindices, nzcnt,
                  beg, varindices, values, effortlevel)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
mcnt	Number of MIP starts to be changed.
mipstartindices	Array of length mcnt containing the numeric indices of the MIP starts to be changed.
nzcnt	Number of entries to be changed.
beg	Array of length mcnt used with varindices and values. Consult the IBM ILOG CPLEX documentation for more detailed information.
varindices	Array of length nzcnt containing the numeric indices of the columns corresponding to the variables which are assigned starting values. Consult the IBM ILOG CPLEX documentation for more detailed information.
values	Array of length nzcnt containing the values to use for the MIP starts. Consult the IBM ILOG CPLEX documentation for more detailed information.
effortlevel	Array of length mcnt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Details

Interface to the C function chgMIPstarts which calls the CPLEX function CPXchg mipstarts.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

chgNameCPLEX	<i>Change the Name of a Constraint a Variable in a CPLEX Problem Object.</i>
--------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgNameCPLEX(env, lp, key, ij, name)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
key	A character to specify whether a row name or a column name should be changed.
ij	An integer that specifies the numeric index of the column or row whose name is to be changed.
name	A pointer to a character string containing the new name.

Details

Interface to the C function chgName which calls the CPLEX function CPXchgname.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`chgObjCPLEX`*Change Linear Objective Coefficients*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgobj. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgObjCPLEX(env, lp, ncols, ind, val)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>ncols</code>	Number of bounds to be changed.
<code>ind</code>	Indices of bounds to be changed.
<code>val</code>	A vector containing the new values of the objective coefficients of the variables specified in <code>ind</code> .

Details

Interface to the C function `chgObj` which calls the CPLEX function `CPXchgobj`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

chgProbNameCPLEX	<i>Change the Name of the Current Problem.</i>
------------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgprobname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgProbNameCPLEX(env, lp, probname)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
probname	The new name of the problem.

Details

Interface to the C function `chgProbName` which calls the CPLEX function `CPXchgprobname`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

chgProbTypeCPLEX *Change the Current Problem to a Related Problem*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgprobtype. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgProbTypeCPLEX(env, lp, ptype)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
ptype	A single integer value specifying the problem type.

Details

Interface to the C function chgProbType which calls the CPLEX function CPXchgprobtype.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[getProbTypeCPLEX](#), [cplexConstants](#) section "Problem Types".

chgQPcoefCPLEX	<i>Change a Single Coefficient in the Quadratic Objective of a Quadratic Problem</i>
----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgqpcoef. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgQPcoefCPLEX(env, lp, i, j, val)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
i	The first variable number.
j	The second variable number.
val	The new coefficient value.

Details

Interface to the C function chgQPcoef which calls the CPLEX function CPXchgqpcoef.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[chgCoefCPLEX](#)

`chgRhsCPLEX`*Change Righthand Side Coefficients*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgrhs. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgRhsCPLEX(env, lp, nrows, ind, val)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>nrows</code>	Number of bounds to be changed.
<code>ind</code>	Indices of bounds to be changed.
<code>val</code>	A vector containing the new values of the objective coefficients of the variables specified in <code>ind</code> .

Details

Interface to the C function `chgRhs` which calls the CPLEX function `CPXchgrhs`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

chgRngValCPLEX *Change Range Coefficients*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgrngval. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgRngValCPLEX(env, lp, nrows, ind, val)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
nrows	Number of bounds to be changed.
ind	Indices of bounds to be changed.
val	A vector containing the new values of the objective coefficients of the variables specified in ind.

Details

Interface to the C function chgRngVal which calls the CPLEX function CPXchgrngval.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

chgRowNameCPLEX *Change Names of Linear Constraints*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgrowname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgRowNameCPLEX(env, lp, nnames, ind, names)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
nnames	A vector that specifies the total number of variable names to be changed.
ind	A vector containing the numeric indices indices of the variables for which the names are to be changed.
names	A vector containing the strings of the new variable names for the columns specified in ind.

Details

Interface to the C function chgRowName which calls the CPLEX function CPXchgrowname.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

`chgSenseCPLEX`*Change Sense of a Set of Linear Constraints*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgsense. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
chgSenseCPLEX(env, lp, nrows, ind, sense)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>nrows</code>	Number of bounds to be changed.
<code>ind</code>	Indices of bounds to be changed.
<code>sense</code>	A vector containing characters that tell the new sense of the linear constraints specified in <code>ind</code> .

Details

Interface to the C function `chgSense` which calls the CPLEX function `CPXchgsense`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

chgTerminateCPLEX *Change Termination Signal*

Description

The function chgTerminateCPLEX changes termination signal.

Usage

```
chgTerminateCPLEX(env, tval = 1)
```

Arguments

env An object of class "cplexPtr" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.

tval Single integer value.

Value

NULL

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[setTerminateCPLEX](#), [delTerminateCPLEX](#), [printTerminateCPLEX](#)

cleanupCoefCPLEX *change Problem Coefficients to Zero That are Smaller in Magnitude Than the Tolerance Specified in the Argument eps*

Description

Low level interface function to the IBM ILOG CPLEX function CPXcleanup. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
cleanupCoefCPLEX(env, lp, eps)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
eps	Single numeric value giving the tolerance.

Details

Interface to the C function `cleanupCoef` which calls the CPLEX function `CPXcleanup`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

cloneProbCPLEX	<i>Copy a CPLEX Problem Object</i>
----------------	------------------------------------

Description

Low level interface function to the IBM ILOG CPLEX function `CPXcloneprob`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
cloneProbCPLEX(env, lp, ptrtype = "cplex_prob")
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
ptrtype	A name for the pointer object.

Details

Interface to the C function `cloneProb` which calls the CPLEX function `CPXcloneprob`.

Value

If successful a pointer to the new CPLEX problem object as returned by `initProbCPLEX` (an object of class "`cpLexPtr`"), otherwise an instance of class "`cpLexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`closeEnvCPLEX`*Free all of the Data Structures Associated With CPLEX*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXcloseCPLEX`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
closeEnvCPLEX(env)
```

Arguments

`env` An object of class "`cpLexPtr`" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function `closeEnv` which calls the CPLEX function `CPXcloseCPLEX`.

Value

Zero if successful, otherwise an instance of class "`cpLexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also[openEnvCPLEX](#)

closeFileCPLEX	<i>Close a File</i>
----------------	---------------------

Description

Low level interface function to the IBM ILOG CPLEX function `CPXfclose`. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where `CPXfclose` has been removed.

Usage

```
closeFileCPLEX(cpfile)
```

Arguments

`cpfile` A pointer to a file as returned by [openFileCPLEX](#).

Details

Interface to the C function `cplexfclose` which calls the CPLEX function `CPXfclose`.

Value

Zero if successful, otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[fileputCPLEX](#), [openFileCPLEX](#)

`closeProbCPLEX`*Close CPLEX Environment And Remove CPLEX Problem Object*

Description

The function `closeProbCPLEX` closes a CPLEX environment and removes a CPLEX problem object.

Usage

```
closeProbCPLEX(prob)
```

Arguments

`prob` A list containing a pointer to an IBM ILOG CPLEX environment and a Pointer to an IBM ILOG CPLEX problem object. Both elements are objects of class "`cpLexPtr`" as returned by `openProbCPLEX`.

Details

Interface to the C functions `delProb` and `closeEnv` calling CPLEX functions `CPXcloseCPLEX` and `CPXfreeprob`.

Value

An integer vector containing the return values of `CPXcloseCPLEX` and `CPXfreeprob`.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[openProbCPLEX](#)

cLpWriteCPLEX	<i>Write an LP Format File Containing Identified Conflict</i>
---------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXclpwrite. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
cLpWriteCPLEX(env, lp, fname)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
fname	Single character value giving the filename to write to.

Details

Interface to the C function `cLpWriteCPLEX` which calls the CPLEX function `CPXclpwrite`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`completeIpCplex`*Manage Modification Steps Closely*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXcompleteIp`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
completeIpCplex(env, lp)
```

Arguments

- | | |
|------------------|---|
| <code>env</code> | An object of class " <code>CplexPtr</code> " as returned by <code>openEnvCplex</code> . This is basically a pointer to an IBM ILOG CPLEX environment. |
| <code>lp</code> | An object of class " <code>CplexPtr</code> " as returned by <code>initProbCplex</code> . This is basically a pointer to an IBM ILOG CPLEX problem object. |

Details

Interface to the C function `completeIpCplex` which calls the CPLEX function `CPXcompleteIp`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`copyBaseCPLEX`*Copies a Basis Into a CPLEX Problem Object.*

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopybase. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
copyBaseCPLEX(env, lp, cstat, rstat)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>cstat</code>	A vector containing the basis status of the columns in the constraint matrix.
<code>rstat</code>	A vector containing the basis status of the slack, or surplus, or artificial variable associated with each row in the constraint matrix.

Details

Interface to the C function `copyBase` which calls the CPLEX function `CPXcopybase`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

copyColTypeCPLEX *Copy Variable Type Information Into a Given Problem*

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgprobname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
copyColTypeCPLEX(env, lp, xtype)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
xctype	A vector containing the type of each column in the constraint matrix.

Details

Interface to the C function `copyColType` which calls the CPLEX function `CPXcopyctype`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

copyLpCPLEX

Copy Data Defining an LP Problem to a CPLEX Problem Object.

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopylp. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
copyLpCPLEX(env, lp, nCols, nRows, lpdire, objf, rhs, sense,
            matbeg, matcnt, matind, matval, lb, ub, rngval = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
nCols	Number of columns in the constraint matrix.
nRows	Number of rows in the constraint matrix.
lpdire	Single integer value that specifies whether the problem is a minimization or maximization problem.
objf	The objective function coefficients.
rhs	The righthand side values for each constraint in the constraint matrix.
sense	The sense of each constraint in the constraint matrix.
matbeg	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matcnt	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
lb	Containing the lower bound on each of the variables.
ub	Containing the lower bound on each of the variables.
rngval	Containing the range value of each ranged constraint.

Details

Interface to the C function copyLp which calls the CPLEX function CPXcopylp.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

copyLpwNamesCPLEX *Copy Data Defining an LP Problem to a CPLEX Problem Object.*

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopylpwnames. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
copyLpwNamesCPLEX(env, lp, nCols, nRows, lpdire, objf, rhs, sense,
                  matbeg, matcnt, matind, matval, lb, ub,
                  rngval = NULL, cnames = NULL, rnames = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
nCols	Number of columns in the constraint matrix.
nRows	Number of rows in the constraint matrix.
lpdire	Single integer value that specifies whether the problem is a minimization or maximization problem.
objf	The objective function coefficients.
rhs	The righthand side values for each constraint in the constraint matrix.
sense	The sense of each constraint in the constraint matrix.
matbeg	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matcnt	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.

matind	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
lb	Containing the lower bound on each of the variables.
ub	Containing the lower bound on each of the variables.
rngval	Containing the range value of each ranged constraint.
cnames	Names of the matrix columns or, equivalently, the variable names.
rnames	Names of the matrix rows or, equivalently, the constraint names.

Details

Interface to the C function copyLpwNames which calls the CPLEX function CPXcopylpwnames.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

copyObjNameCPLEX	<i>Copy a Name for the Objective Function Into a CPLEX Problem Object.</i>
------------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopyobjname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
copyObjNameCPLEX(env, lp, oname)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
oname	A pointer to a character string containing the objective name.

Details

Interface to the C function copyObjName which calls the CPLEX function CPXcopyobjname.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

copyOrderCPLEX

Copy Priority Order to CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopyorder. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
copyOrderCPLEX(env, lp, cnt, indices, priority = NULL, direction = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
cnt	Number of entries.
indices	Indices of the columns corresponding to the integer variables that are assigned priorities.
priority	Priorities assigned to the integer variables.
direction	Branching direction assigned to the integer variables.

Details

Interface to the C function copyOrder which calls the CPLEX function CPXcopyorder.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

copyQPsepCPLEX

Copy the Quadratic Objective Matrix Q for a Separable QP Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopyqpsep. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
copyQPsepCPLEX(env, lp, qsepvec)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
qsepvec	A vector containing the quadratic coefficients.

Details

Interface to the C function copyQPsep which calls the CPLEX function CPXcopyqpsep.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`copyQuadCPLEX`*Copy a Quadratic Objective Matrix Q When Q is not Diagonal.*

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopyquad. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
copyQuadCPLEX(env, lp, qmatbeg, qmatcnt, qmatind, qmatval)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>qmatbeg</code>	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
<code>qmatcnt</code>	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
<code>qmatind</code>	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
<code>qmatval</code>	Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.

Details

Interface to the C function `copyQuad` which calls the CPLEX function `CPXcopyquad`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

copyStartCPLEX	<i>Provides Starting Information for Use in a Subsequent Call to a Simplex Optimization Routine.</i>
----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopystart. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
copyStartCPLEX(env, lp,
               cstat = NULL, rstat = NULL,
               cprim = NULL, rprim = NULL,
               cdual = NULL, rdual = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
cstat	A vector containing the basis status of the columns in the constraint matrix.
rstat	A vector containing the basis status of the slack, surplus, or artificial variable associated with each row in the constraint matrix.
cprim	A vector containing the initial primal values of the column variables.
rprim	A vector containing the initial primal values of the slack (row) variables.
cdual	A vector containing the initial values of the reduced costs for the column variables.
rdual	A vector containing the initial values of the dual variables for the rows.

Details

Interface to the C function `copyStart` which calls the CPLEX function `CPXcopyStart`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
 Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

cplexConstants *Constants, Parameters, Return and Status Codes of IBM ILOG CPLEX*

Description

This is a list containing constants used by IBM ILOG CPLEX. Consult the IBM ILOG CPLEX manual for more information, in particular for the use of control parameters.

General Parameters

CPX_INFBOUND	1.0E+20
CPX_STR_PARAM_MAX	512

Types of parameters

CPX_PARAMTYPE_NONE	0
CPX_PARAMTYPE_INT	1
CPX_PARAMTYPE_DOUBLE	2
CPX_PARAMTYPE_STRING	3
CPX_PARAMTYPE_LONG	4

Values returned for stat by solution

CPX_STAT_OPTIMAL	1
CPX_STAT_UNBOUNDED	2
CPX_STAT_INFEASIBLE	3
CPX_STAT_INFOrUNBD	4
CPX_STAT_OPTIMAL_INFEAS	5
CPX_STAT_NUM_BEST	6
CPX_STAT_ABORT_IT_LIM	10
CPX_STAT_ABORT_TIME_LIM	11
CPX_STAT_ABORT_OBJ_LIM	12
CPX_STAT_ABORT_USER	13
CPX_STAT_FEASIBLE_RELAXED_SUM	14
CPX_STAT_OPTIMAL_RELAXED_SUM	15
CPX_STAT_FEASIBLE_RELAXED_INF	16
CPX_STAT_OPTIMAL_RELAXED_INF	17
CPX_STAT_FEASIBLE_RELAXED_QUAD	18

CPX_STAT_OPTIMAL_RELAXED_QUAD	19
CPX_STAT_FEASIBLE	23
CPX_STAT_ABORT_DETTIME_LIM	25

Solution type return values from CPXsolninfo

CPX_NO_SOLN	0
CPX_BASIC_SOLN	1
CPX_NONBASIC_SOLN	2
CPX_PRIMAL_SOLN	3

Values of presolve stats for columns and rows

CPX_PRECOL_LOW	-1	fixed to original lb
CPX_PRECOL_UP	-2	fixed to original ub
CPX_PRECOL_FIX	-3	fixed to some other value
CPX_PRECOL_AGG	-4	aggregated $y = a \times x + b$
CPX_PRECOL_OTHER	-5	cannot be expressed by a linear combination of active variables in the presolved model crushing w
CPX_PREROW_RED	-1	redundant row removed in presolved model
CPX_PREROW_AGG	-2	used to aggregate a variable
CPX_PREROW_OTHER	-3	other, for example merge two inequalities into a single equation

Generic constants

CPX_ON	1
CPX_OFF	0
CPX_MAX	-1
CPX_MIN	1

Primal simplex pricing algorithm

CPX_PPRIIND_PARTIAL	-1
CPX_PPRIIND_AUTO	0
CPX_PPRIIND_DEVEX	1

CPX_PPRIIND_STEEP	2
CPX_PPRIIND_STEEPQSTART	3
CPX_PPRIIND_FULL	4

Dual simplex pricing algorithm

CPX_DPRIIND_AUTO	0
CPX_DPRIIND_FULL	1
CPX_DPRIIND_STEEP	2
CPX_DPRIIND_FULL_STEEP	3
CPX_DPRIIND_STEEPQSTART	4
CPX_DPRIIND_DEVEX	5

PARALLELMODE values

CPX_PARALLEL_DETERMINISTIC	1
CPX_PARALLEL_AUTO	0
CPX_PARALLEL_OPPORTUNISTIC	-1

Values for CPX_PARAM_WRITELEVEL

CPX_WRITELEVEL_AUTO	0
CPX_WRITELEVEL_ALLVARS	1
CPX_WRITELEVEL_DISCRETEVARS	2
CPX_WRITELEVEL_NONZEROVARS	3
CPX_WRITELEVEL_NONZERODISCRETEVARS	4

Values for CPX_PARAM_SOLUTIONTARGET

CPX_SOLUTIONTARGET_AUTO	0
CPX_SOLUTIONTARGET_OPTIMALCONVEX	1
CPX_SOLUTIONTARGET_FIRSTORDER	2
CPX_SOLUTIONTARGET_OPTIMALGLOBAL	3

LP/QP solution algorithms

Used as possible values for CPX_PARAM_LPMETHOD, CPX_PARAM_QPMETHOD, CPX_PARAM_BARCROSSALG, CPXgetmethod, ...

CPX_ALG_NONE	-1
CPX_ALG_AUTOMATIC	0
CPX_ALG_PRIMAL	1
CPX_ALG_DUAL	2
CPX_ALG_NET	3
CPX_ALG_BARRIER	4
CPX_ALG_SIFTING	5
CPX_ALG_CONCURRENT	6
CPX_ALG_BAROPT	7
CPX_ALG_PIVOTIN	8
CPX_ALG_PIVOTOUT	9
CPX_ALG_PIVOT	10
CPX_ALG_FEASOPT	11
CPX_ALG_MIP	12
CPX_ALG_ROBUST	13

Basis status values

CPX_AT_LOWER	0
CPX_BASIC	1
CPX_AT_UPPER	2
CPX_FREE_SUPER	3

Variable types for ctype array

CPX_CONTINUOUS	"C"
CPX_BINARY	"B"
CPX_INTEGER	"I"
CPX_SEMICONT	"S"
CPX_SEMIINT	"N"

PREREDUCE settings

CPX_PREREDUCE_PRIMALANDDUAL	3
CPX_PREREDUCE_DUALONLY	2
CPX_PREREDUCE_PRIMALONLY	1
CPX_PREREDUCE_NOPRIMALORDUAL	0

Conflict statuses

CPX_STAT_CONFLICT_FEASIBLE	30
CPX_STAT_CONFLICT_MINIMAL	31
CPX_STAT_CONFLICT_ABORT_CONTRADICTION	32
CPX_STAT_CONFLICT_ABORT_TIME_LIM	33
CPX_STAT_CONFLICT_ABORT_IT_LIM	34
CPX_STAT_CONFLICT_ABORT_NODE_LIM	35
CPX_STAT_CONFLICT_ABORT_OBJ_LIM	36
CPX_STAT_CONFLICT_ABORT_MEM_LIM	37
CPX_STAT_CONFLICT_ABORT_USER	38
CPX_STAT_CONFLICT_ABORT_DETTIME_LIM	39

Conflict status values

CPX_CONFLICT_EXCLUDED	-1
CPX_CONFLICT_POSSIBLE_MEMBER	0
CPX_CONFLICT_POSSIBLE_LB	1
CPX_CONFLICT_POSSIBLE_UB	2
CPX_CONFLICT_MEMBER	3
CPX_CONFLICT_LB	4
CPX_CONFLICT_UB	5

Problem Types

Types 4, 9, and 12 are internal, the others are for users.

CPXPROB_LP	0
CPXPROB_MILP	1
CPXPROB_FIXEDMILP	3
CPXPROB_NODELP	4
CPXPROB_QP	5
CPXPROB_MIQP	7

CPXPROB_FIXEDMIQP	8
CPXPROB_NODEQP	9
CPXPROB_QCP	10
CPXPROB_MIQCP	11
CPXPROB_NODEQCP	12

CPLEX Parameter numbers

CPX_PARAM_ADVIND	1001
CPX_PARAM_AGGFILL	1002
CPX_PARAM_AGGIND	1003
CPX_PARAM_BASINTERVAL	1004
CPX_PARAM_CFILEMUL	1005
CPX_PARAM_CLOCKTYPE	1006
CPX_PARAM_CRAIND	1007
CPX_PARAM_DEPIND	1008
CPX_PARAM_DPRIIND	1009
CPX_PARAM_PRICELIM	1010
CPX_PARAM_EPMRK	1013
CPX_PARAM_EPOPT	1014
CPX_PARAM_EPPER	1015
CPX_PARAM_EPRHS	1016
CPX_PARAM_FASTMIP	1017
CPX_PARAM_SIMDISPLAY	1019
CPX_PARAM_ITLIM	1020
CPX_PARAM_ROWREADLIM	1021
CPX_PARAM_NETFIND	1022
CPX_PARAM_COLREADLIM	1023
CPX_PARAM_NZREADLIM	1024
CPX_PARAM_OBJLLIM	1025
CPX_PARAM_OBJULIM	1026
CPX_PARAM_PERIND	1027
CPX_PARAM_PERLIM	1028
CPX_PARAM_PPRIIND	1029
CPX_PARAM_PREIND	1030
CPX_PARAM_REINV	1031
CPX_PARAM_REVERSEIND	1032
CPX_PARAM_RFILEMUL	1033
CPX_PARAM_SCAIND	1034
CPX_PARAM_SCRIND	1035
CPX_PARAM_SINGLIM	1037
CPX_PARAM_SINGTOL	1038
CPX_PARAM_TILIM	1039
CPX_PARAM_XXXIND	1041
CPX_PARAM_PREDUAL	1044
CPX_PARAM_EPOPT_H	1049
CPX_PARAM_EPRHS_H	1050

CPX_PARAM_PREPASS	1052
CPX_PARAM_DATACHECK	1056
CPX_PARAM_REDUCE	1057
CPX_PARAM_PRELINEAR	1058
CPX_PARAM_LPMETHOD	1062
CPX_PARAM_QPMETHOD	1063
CPX_PARAM_WORKDIR	1064
CPX_PARAM_WORKMEM	1065
CPX_PARAM_THREADS	1067
CPX_PARAM_CONFLICTDISPLAY	1074
CPX_PARAM_SIFTDISPLAY	1076
CPX_PARAM_SIFTALG	1077
CPX_PARAM_SIFTITLIM	1078
CPX_PARAM_MPSSLONGNUM	1081
CPX_PARAM_MEMORYEMPHASIS	1082
CPX_PARAM_NUMERICALEMPHASIS	1083
CPX_PARAM_FEASOPTMODE	1084
CPX_PARAM_PARALLELMODE	1109
CPX_PARAM_TUNINGMEASURE	1110
CPX_PARAM_TUNINGREPEAT	1111
CPX_PARAM_TUNINGTILIM	1112
CPX_PARAM_TUNINGDISPLAY	1113
CPX_PARAM_WRITELEVEL	1114
CPX_PARAM_RANDOMSEED	1124
CPX_PARAM_DETTILIM	1127
CPX_PARAM_FILEENCODING	1129
CPX_PARAM_APIENCODING	1130
CPX_PARAM_SOLUTIONTARGET	1131
CPX_PARAM_CLONELOG	1132
CPX_PARAM_TUNINGDETTILIM	1139
CPX_PARAM_ALL_MIN	1000
CPX_PARAM_ALL_MAX	6000

Values for CPX_PARAM_TUNINGMEASURE

CPX_TUNE_AVERAGE	1
CPX_TUNE_MINMAX	2

Values for incomplete tuning

CPX_TUNE_ABORT	1
----------------	---

CPX_TUNE_TILIM	2
CPX_TUNE_DETTILIM	3

Quality query identifiers

CPX_MAX_PRIMAL_INFEAS	1
CPX_MAX_SCALED_PRIMAL_INFEAS	2
CPX_SUM_PRIMAL_INFEAS	3
CPX_SUM_SCALED_PRIMAL_INFEAS	4
CPX_MAX_DUAL_INFEAS	5
CPX_MAX_SCALED_DUAL_INFEAS	6
CPX_SUM_DUAL_INFEAS	7
CPX_SUM_SCALED_DUAL_INFEAS	8
CPX_MAX_INT_INFEAS	9
CPX_SUM_INT_INFEAS	10
CPX_MAX_PRIMAL_RESIDUAL	11
CPX_MAX_SCALED_PRIMAL_RESIDUAL	12
CPX_SUM_PRIMAL_RESIDUAL	13
CPX_SUM_SCALED_PRIMAL_RESIDUAL	14
CPX_MAX_DUAL_RESIDUAL	15
CPX_MAX_SCALED_DUAL_RESIDUAL	16
CPX_SUM_DUAL_RESIDUAL	17
CPX_SUM_SCALED_DUAL_RESIDUAL	18
CPX_MAX_COMP_SLACK	19
CPX_SUM_COMP_SLACK	21
CPX_MAX_X	23
CPX_MAX_SCALED_X	24
CPX_MAX_PI	25
CPX_MAX_SCALED_PI	26
CPX_MAX_SLACK	27
CPX_MAX_SCALED_SLACK	28
CPX_MAX_RED_COST	29
CPX_MAX_SCALED_RED_COST	30
CPX_SUM_X	31
CPX_SUM_SCALED_X	32
CPX_SUM_PI	33
CPX_SUM_SCALED_PI	34
CPX_SUM_SLACK	35
CPX_SUM_SCALED_SLACK	36
CPX_SUM_RED_COST	37
CPX_SUM_SCALED_RED_COST	38
CPX_KAPPA	39
CPX_OBJ_GAP	40
CPX_DUAL_OBJ	41
CPX_PRIMAL_OBJ	42

CPX_MAX_QCPRIMAL_RESIDUAL	43
CPX_SUM_QCPRIMAL_RESIDUAL	44
CPX_MAX_QCSLACK_INFEAS	45
CPX_SUM_QCSLACK_INFEAS	46
CPX_MAX_QCSLACK	47
CPX_SUM_QCSLACK	48
CPX_MAX_INDSLACK_INFEAS	49
CPX_SUM_INDSLACK_INFEAS	50
CPX_EXACT_KAPPA	51
CPX_KAPPA_STABLE	52
CPX_KAPPA_SUSPICIOUS	53
CPX_KAPPA_UNSTABLE	54
CPX_KAPPA_ILLPOSED	55
CPX_KAPPA_MAX	56
CPX_KAPPA_ATTENTION	57

Solution quality symbols new in CPLEX 12.9.0

CPX_MAX_PWLSLACK_INFEAS	58
CPX_SUM_PWLSLACK_INFEAS	59

feasopt options

CPX_FEASOPT_MIN_SUM	0
CPX_FEASOPT_OPT_SUM	1
CPX_FEASOPT_MIN_INF	2
CPX_FEASOPT_OPT_INF	3
CPX_FEASOPT_MIN_QUAD	4
CPX_FEASOPT_OPT_QUAD	5

File: barconst.h

CPX_STAT_OPTIMAL_FACE_UNBOUNDED	20
CPX_STAT_ABORT_PRIM_OBJ_LIM	21
CPX_STAT_ABORT_DUAL_OBJ_LIM	22
CPX_STAT_FIRSTORDER	24

Barrier parameters

CPX_PARAM_BARDSTART	3001
CPX_PARAM_BAREPCOMP	3002
CPX_PARAM_BARGROWTH	3003
CPX_PARAM_BAROBJRNG	3004
CPX_PARAM_BARPSTART	3005
CPX_PARAM_BARALG	3007
CPX_PARAM_BARCOLNZ	3009
CPX_PARAM_BARDISPLAY	3010
CPX_PARAM_BARITLIM	3012
CPX_PARAM_BARMAXCOR	3013
CPX_PARAM_BARORDER	3014
CPX_PARAM_BARSTARTALG	3017
CPX_PARAM_BARCROSSALG	3018
CPX_PARAM_BARQCPEPCOMP	3020

Optimizing Problems

CPX_BARORDER_AUTO	0
CPX_BARORDER_AMD	1
CPX_BARORDER_AMF	2
CPX_BARORDER_ND	3

MIP emphasis settings

CPX_MIPEMPHASIS_BALANCED	0
CPX_MIPEMPHASIS_FEASIBILITY	1
CPX_MIPEMPHASIS_OPTIMALITY	2
CPX_MIPEMPHASIS_BESTBOUND	3
CPX_MIPEMPHASIS_HIDDENFEAS	4

Values for sostype and branch type

CPX_TYPE_VAR	"0"
--------------	-----

CPX_TYPE_SOS1	"1"
CPX_TYPE_SOS2	"2"
CPX_TYPE_USER	"X"
CPX_TYPE_ANY	"A"

Variable selection values

CPX_VARSEL_MININFEAS	-1
CPX_VARSEL_DEFAULT	0
CPX_VARSEL_MAXINFEAS	1
CPX_VARSEL_PSEUDO	2
CPX_VARSEL_STRONG	3
CPX_VARSEL_PSEUDOREduced	4

Node selection values

CPX_NODESEL_DFS	0
CPX_NODESEL_BESTBOUND	1
CPX_NODESEL_BESTEST	2
CPX_NODESEL_BESTEST_ALT	3

Values for generated priority order

CPX_MIPORDER_COST	1
CPX_MIPORDER_BOUNDS	2
CPX_MIPORDER_SCALED_COST	3

Values for direction array

CPX_BRANCH_GLOBAL	0
CPX_BRANCH_DOWN	-1
CPX_BRANCH_UP	1

Values for CPX_PARAM_BRDIR

CPX_BRDIR_DOWN	-1
CPX_BRDIR_AUTO	0
CPX_BRDIR_UP	1

Values for CPX_PARAM_MIPSEARCH

CPX_MIPSEARCH_AUTO	0
CPX_MIPSEARCH_TRADITIONAL	1
CPX_MIPSEARCH_DYNAMIC	2

Values for CPX_PARAM_MIPKAPPASTATS

CPX_MIPKAPPA_OFF	-1
CPX_MIPKAPPA_AUTO	0
CPX_MIPKAPPA_SAMPLE	1
CPX_MIPKAPPA_FULL	2

Effort levels for MIP starts

CPX_MIPSTART_AUTO	0
CPX_MIPSTART_CHECKFEAS	1
CPX_MIPSTART_SOLVEMIP	2
CPX_MIPSTART_SOLVEMIP	3
CPX_MIPSTART_REPAIR	4

MIP Problem status codes

CPXMIP_OPTIMAL	101
CPXMIP_OPTIMAL_TOL	102

CPXMIP_INFEASIBLE	103
CPXMIP_SOL_LIM	104
CPXMIP_NODE_LIM_FEAS	105
CPXMIP_NODE_LIM_INFEAS	106
CPXMIP_TIME_LIM_FEAS	107
CPXMIP_TIME_LIM_INFEAS	108
CPXMIP_FAIL_FEAS	109
CPXMIP_FAIL_INFEAS	110
CPXMIP_MEM_LIM_FEAS	111
CPXMIP_MEM_LIM_INFEAS	112
CPXMIP_ABORT_FEAS	113
CPXMIP_ABORT_INFEAS	114
CPXMIP_OPTIMAL_INFEAS	115
CPXMIP_FAIL_FEAS_NO_TREE	116
CPXMIP_FAIL_INFEAS_NO_TREE	117
CPXMIP_UNBOUNDED	118
CPXMIP_INForUNBD	119
CPXMIP_FEASIBLE_RELAXED_SUM	120
CPXMIP_OPTIMAL_RELAXED_SUM	121
CPXMIP_FEASIBLE_RELAXED_INF	122
CPXMIP_OPTIMAL_RELAXED_INF	123
CPXMIP_FEASIBLE_RELAXED_QUAD	124
CPXMIP_OPTIMAL_RELAXED_QUAD	125
CPXMIP_ABORT_RELAXED	126
CPXMIP_FEASIBLE	127
CPXMIP_POPULATESOL_LIM	128
CPXMIP_OPTIMAL_POPULATED	129
CPXMIP_OPTIMAL_POPULATED_TOL	130
CPXMIP_DETTIME_LIM_FEAS	131
CPXMIP_DETTIME_LIM_INFEAS	132

Valid purgeable values for adding usercuts and lazyconstraints

CPX_USECUT_FORCE	0
CPX_USECUT_PURGE	1
CPX_USECUT_FILTER	2

For CPXgetnodeintfeas

CPX_INTEGER_FEASIBLE	0
CPX_INTEGER_INFEASIBLE	1

CPX_IMPLIED_INTEGER_FEASIBLE 2

MIP Parameter numbers

CPX_PARAM_BRDIR	2001
CPX_PARAM_BTTOL	2002
CPX_PARAM_CLIQUES	2003
CPX_PARAM_COEREDIND	2004
CPX_PARAM_COVERS	2005
CPX_PARAM_CUTLO	2006
CPX_PARAM_CUTUP	2007
CPX_PARAM_EPAGAP	2008
CPX_PARAM_EPGAP	2009
CPX_PARAM_EPINT	2010
CPX_PARAM_MIPDISPLAY	2012
CPX_PARAM_MIPINTERVAL	2013
CPX_PARAM_INTSOLLIM	2015
CPX_PARAM_NODEFILEIND	2016
CPX_PARAM_NODELIM	2017
CPX_PARAM_NODESEL	2018
CPX_PARAM_OBJDIF	2019
CPX_PARAM_MIPORDIND	2020
CPX_PARAM_RELOBJDIF	2022
CPX_PARAM_STARTALG	2025
CPX_PARAM_SUBALG	2026
CPX_PARAM_TRELIM	2027
CPX_PARAM_VARSEL	2028
CPX_PARAM_BNDSTRENIND	2029
CPX_PARAM_HEURFREQ	2031
CPX_PARAM_MIPORDTYPE	2032
CPX_PARAM_CUTSFACOR	2033
CPX_PARAM_RELAXPREIND	2034
CPX_PARAM_PRESLVND	2037
CPX_PARAM_BBINTERVAL	2039
CPX_PARAM_FLOWCOVERS	2040
CPX_PARAM_IMPLBD	2041
CPX_PARAM_PROBE	2042
CPX_PARAM_GUBCOVERS	2044
CPX_PARAM_STRONGCANDLIM	2045
CPX_PARAM_STRONGITLIM	2046
CPX_PARAM_FRACCAND	2048
CPX_PARAM_FRACCUTS	2049
CPX_PARAM_FRACPASS	2050
CPX_PARAM_FLOWPATHS	2051
CPX_PARAM_MIRCUTS	2052

CPX_PARAM_DISJCUTS	2053
CPX_PARAM_AGGCUTLIM	2054
CPX_PARAM_MIPCBREDLP	2055
CPX_PARAM_CUTPASS	2056
CPX_PARAM_MIPEMPHASIS	2058
CPX_PARAM_SYMMETRY	2059
CPX_PARAM_DIVETYPE	2060
CPX_PARAM_RINSHEUR	2061
CPX_PARAM_SUBMIPNODELIM	2062
CPX_PARAM_LBHEUR	2063
CPX_PARAM_REPEATPRESOLVE	2064
CPX_PARAM_PROBETIME	2065
CPX_PARAM_POLISHTIME	2066
CPX_PARAM_REPAIRTRIES	2067
CPX_PARAM_EPLIN	2068
CPX_PARAM_EPRELAX	2073
CPX_PARAM_FPHEUR	2098
CPX_PARAM_EACHCUTLIM	2102
CPX_PARAM_SOLNPOOLCAPACITY	2103
CPX_PARAM_SOLNPOOLREPLACE	2104
CPX_PARAM_SOLNPOOLGAP	2105
CPX_PARAM_SOLNPOOLAGAP	2106
CPX_PARAM_SOLNPOOLINTENSITY	2107
CPX_PARAM_POPULATELIM	2108
CPX_PARAM_MIPSEARCH	2109
CPX_PARAM_MIQCPSTRAT	2110
CPX_PARAM_ZEROHALFCUTS	2111
CPX_PARAM_POLISHAFTEREPAGAP	2126
CPX_PARAM_POLISHAFTEREPGAP	2127
CPX_PARAM_POLISHAFTERNODE	2128
CPX_PARAM_POLISHAFTERINTSOL	2129
CPX_PARAM_POLISHAFTERTIME	2130
CPX_PARAM_MCFCUTS	2134
CPX_PARAM_MIPKAPPASTATS	2137
CPX_PARAM_AUXROOTTHREADS	2139
CPX_PARAM_INTSOLFILEPREFIX	2143
CPX_PARAM_PROBEDETTIME	2150
CPX_PARAM_POLISHAFTERDETTIME	2151
CPX_PARAM_LANDPCUTS	2152
CPX_PARAM_RAMPUPDURATION	2163
CPX_PARAM_RAMPUPDETTILIM	2164
CPX_PARAM_RAMPUPUTILIM	2165

Values for CPX_PARAM_SOLNPOOLREPLACE

CPX_SOLNPOOL_FIFO	0
CPX_SOLNPOOL_OBJ	1
CPX_SOLNPOOL_DIV	2
CPX_SOLNPOOL_FILTER_DIVERSITY	1
CPX_SOLNPOOL_FILTER_RANGE	2

File: gconst.h

CPX_CON_LOWER_BOUND	1
CPX_CON_UPPER_BOUND	2
CPX_CON_LINEAR	3
CPX_CON_QUADRATIC	4
CPX_CON_SOS	5
CPX_CON_INDICATOR	6

internal types

CPX_CON_MINEXPR	7
CPX_CON_MAXEXPR	8
CPX_CON_PWL	9
CPX_CON_ABS	9
CPX_CON_DISJCST	10
CPX_CON_INDDISJCST	11
CPX_CON_SETVAR	12
CPX_CON_SETVARMEMBER	13
CPX_CON_SETVARCARD	14
CPX_CON_SETVARSUM	15
CPX_CON_SETVARMIN	16
CPX_CON_SETVARMAX	17
CPX_CON_SETVARSUBSET	18
CPX_CON_SETVARDOMAIN	19
CPX_CON_SETVARUNION	20
CPX_CON_SETVARINTERSECTION	21
CPX_CON_SETVARNULLINTERSECT	22
CPX_CON_SETVARINTERSECT	23
CPX_CON_SETVAREQ	24
CPX_CON_SETVARNEQ	25
CPX_CON_SETVARNEQCST	26
CPX_CON_LAST_CONTYPE	27

Network parameters

CPX_PARAM_NETITLIM	5001
CPX_PARAM_NETEPOPT	5002
CPX_PARAM_NETEPRHS	5003
CPX_PARAM_NETPPRIIND	5004
CPX_PARAM_NETDISPLAY	5005

NETOPT display values

CPXNET_NO_DISPLAY_OBJECTIVE	0
CPXNET_TRUE_OBJECTIVE	1
CPXNET_PENALIZED_OBJECTIVE	2

NETOPT pricing parameters

CPXNET_PRICE_AUTO	0
CPXNET_PRICE_PARTIAL	1
CPXNET_PRICE_MULT_PART	2
CPXNET_PRICE_SORT_MULT_PART	3

Copying data

CPX_PARAM_QPNZREADLIM	4001
-----------------------	------

Specify how to calculate duals for QCPs

CPX_PARAM_CALCQCPDUALS	4003
------------------------	------

presolve

CPX_PARAM_QPMAKEPSDIND	4010
------------------------	------

Error codes

Callable library miscellaneous routines

CPXERR_NEGATIVE_SURPLUS	1207
CPXERR_NO_SENSIT	1260

Error codes new in CPLEX 12.8.0

Callable library miscellaneous routines

CPXERR_CALLBACK_INCONSISTENT	1060
CPXERR_CAND_NOT_RAY	3026
CPXERR_CAND_NOT_POINT	3025

Error codes new in CPLEX 12.9.0

Callable library miscellaneous routines

CPXERR_BAD_MULTIOBJ_ATTR	1488
CPXERR_MULTIOBJ_SUBPROB_SOLVE	1300
CPXERR_NO_OBJ_NAME	1486
CPXERR_NOT_FOR_MULTIOBJ	1070

new parameter names introduced in IBM ILOG CPLEX version 12.6

Callable library miscellaneous routines

CPXPARAM_Advance	1001
CPXPARAM_Barrier_Algorithm	3007
CPXPARAM_Barrier_ColNonzeros	3009
CPXPARAM_Barrier_ConvergeTol	3002
CPXPARAM_Barrier_Crossover	3018
CPXPARAM_Barrier_Display	3010
CPXPARAM_Barrier_Limits_Corrections	3013
CPXPARAM_Barrier_Limits_Growth	3003
CPXPARAM_Barrier_Limits_Iteration	3012
CPXPARAM_Barrier_Limits_ObjRange	3004
CPXPARAM_Barrier_Ordering	3014

CPXPARAM_Barrier_QCPConvergeTol	3020
CPXPARAM_Barrier_StartAlg	3017
CPXPARAM_ClockType	1006
CPXPARAM_Conflict_Display	1074
CPXPARAM_DetTimeLimit	1127
CPXPARAM_DistMIP_Rampup_DetTimeLimit	2164
CPXPARAM_DistMIP_Rampup_Duration	2163
CPXPARAM_DistMIP_Rampup_TimeLimit	2165
CPXPARAM_Emphasis_Memory	1082
CPXPARAM_Emphasis_MIP	2058
CPXPARAM_Emphasis_Numerical	1083
CPXPARAM_Feasopt_Mode	1084
CPXPARAM_Feasopt_Tolerance	2073
CPXPARAM_LPMethod	1062
CPXPARAM_MIP_Cuts_Cliques	2003
CPXPARAM_MIP_Cuts_Covers	2005
CPXPARAM_MIP_Cuts_Disjunctive	2053
CPXPARAM_MIP_Cuts_FlowCovers	2040
CPXPARAM_MIP_Cuts_Gomory	2049
CPXPARAM_MIP_Cuts_GUBCovers	2044
CPXPARAM_MIP_Cuts_Implied	2041
CPXPARAM_MIP_Cuts_LiftProj	2152
CPXPARAM_MIP_Cuts_MCFCut	2134
CPXPARAM_MIP_Cuts_MIRCut	2052
CPXPARAM_MIP_Cuts_PathCut	2051
CPXPARAM_MIP_Cuts_ZeroHalfCut	2111
CPXPARAM_MIP_Display	2012
CPXPARAM_MIP_Interval	2013
CPXPARAM_MIP_Limits_AggForCut	2054
CPXPARAM_MIP_Limits_AuxRootThreads	2139
CPXPARAM_MIP_Limits_CutPasses	2056
CPXPARAM_MIP_Limits_CutsFactor	2033
CPXPARAM_MIP_Limits_EachCutLimit	2102
CPXPARAM_MIP_Limits_GomoryCand	2048
CPXPARAM_MIP_Limits_GomoryPass	2050
CPXPARAM_MIP_Limits_Nodes	2017
CPXPARAM_MIP_Limits_PolishTime	2066
CPXPARAM_MIP_Limits_Populate	2108
CPXPARAM_MIP_Limits_ProbeDetTime	2150
CPXPARAM_MIP_Limits_ProbeTime	2065
CPXPARAM_MIP_Limits_RepairTries	2067
CPXPARAM_MIP_Limits_Solutions	2015
CPXPARAM_MIP_Limits_StrongCand	2045
CPXPARAM_MIP_Limits_StrongIt	2046
CPXPARAM_MIP_Limits_SubMIPNodeLim	2062
CPXPARAM_MIP_Limits_TreeMemory	2027
CPXPARAM_MIP_OrderType	2032
CPXPARAM_MIP_PolishAfter_AbsMIPGap	2126

CPXPARAM_MIP_PolishAfter_DetTime	2151
CPXPARAM_MIP_PolishAfter_MIPGap	2127
CPXPARAM_MIP_PolishAfter_Nodes	2128
CPXPARAM_MIP_PolishAfter_Solutions	2129
CPXPARAM_MIP_PolishAfter_Time	2130
CPXPARAM_MIP_Pool_AbsGap	2106
CPXPARAM_MIP_Pool_Capacity	2103
CPXPARAM_MIP_Pool_Intensity	2107
CPXPARAM_MIP_Pool_RelGap	2105
CPXPARAM_MIP_Pool_Replace	2104
CPXPARAM_MIP_Strategy_Backtrack	2002
CPXPARAM_MIP_Strategy_BBInterval	2039
CPXPARAM_MIP_Strategy_Branch	2001
CPXPARAM_MIP_Strategy_CallbackReducedLP	2055
CPXPARAM_MIP_Strategy_Dive	2060
CPXPARAM_MIP_Strategy_File	2016
CPXPARAM_MIP_Strategy_FPHeur	2098
CPXPARAM_MIP_Strategy_HeuristicFreq	2031
CPXPARAM_MIP_Strategy_KappaStats	2137
CPXPARAM_MIP_Strategy_LBHeur	2063
CPXPARAM_MIP_Strategy_MIQCPStrat	2110
CPXPARAM_MIP_Strategy_NodeSelect	2018
CPXPARAM_MIP_Strategy_Order	2020
CPXPARAM_MIP_Strategy_PresolveNode	2037
CPXPARAM_MIP_Strategy_Probe	2042
CPXPARAM_MIP_Strategy_RINSHeur	2061
CPXPARAM_MIP_Strategy_Search	2109
CPXPARAM_MIP_Strategy_StartAlgorithm	2025
CPXPARAM_MIP_Strategy_SubAlgorithm	2026
CPXPARAM_MIP_Strategy_VariableSelect	2028
CPXPARAM_MIP_Tolerances_AbsMIPGap	2008
CPXPARAM_MIP_Tolerances_Integrality	2010
CPXPARAM_MIP_Tolerances_LowerCutoff	2006
CPXPARAM_MIP_Tolerances_MIPGap	2009
CPXPARAM_MIP_Tolerances_ObjDifference	2019
CPXPARAM_MIP_Tolerances_RelObjDifference	2022
CPXPARAM_MIP_Tolerances_UpperCutoff	2007
CPXPARAM_Network_Display	5005
CPXPARAM_Network_Iterations	5001
CPXPARAM_Network_NetFind	1022
CPXPARAM_Network_Pricing	5004
CPXPARAM_Network_Tolerances_Feasibility	5003
CPXPARAM_Network_Tolerances_Optimality	5002
CPXPARAM_Output_CloneLog	1132
CPXPARAM_Output_IntSolFilePrefix	2143
CPXPARAM_Output_MPSPong	1081
CPXPARAM_Output_WriteLevel	1114
CPXPARAM_Parallel	1109

CPXPARAM_Preprocessing_Aggregator	1003
CPXPARAM_Preprocessing_BoundStrength	2029
CPXPARAM_Preprocessing_CoeffReduce	2004
CPXPARAM_Preprocessing_Dependency	1008
CPXPARAM_Preprocessing_Dual	1044
CPXPARAM_Preprocessing_Fill	1002
CPXPARAM_Preprocessing_Linear	1058
CPXPARAM_Preprocessing_NumPass	1052
CPXPARAM_Preprocessing_Presolve	1030
CPXPARAM_Preprocessing_QCPDuals	4003
CPXPARAM_Preprocessing_QPMakePSD	4010
CPXPARAM_Preprocessing_Reduce	1057
CPXPARAM_Preprocessing_Relax	2034
CPXPARAM_Preprocessing_RepeatPresolve	2064
CPXPARAM_Preprocessing_Symmetry	2059
CPXPARAM_QPMethod	1063
CPXPARAM_RandomSeed	1124
CPXPARAM_Read_APIEncoding	1130
CPXPARAM_Read_Constraints	1021
CPXPARAM_Read_DataCheck	1056
CPXPARAM_Read_FileEncoding	1129
CPXPARAM_Read_Nonzeros	1024
CPXPARAM_Read_QPNonzeros	4001
CPXPARAM_Read_Scale	1034
CPXPARAM_Read_Variables	1023
CPXPARAM_ScreenOutput	1035
CPXPARAM_Sifting_Algorithm	1077
CPXPARAM_Sifting_Display	1076
CPXPARAM_Sifting_Iterations	1078
CPXPARAM_Simplex_Crash	1007
CPXPARAM_Simplex_DGradient	1009
CPXPARAM_Simplex_Display	1019
CPXPARAM_Simplex_Limits_Iterations	1020
CPXPARAM_Simplex_Limits_LowerObj	1025
CPXPARAM_Simplex_Limits_Perturbation	1028
CPXPARAM_Simplex_Limits_Singularity	1037
CPXPARAM_Simplex_Limits_UpperObj	1026
CPXPARAM_Simplex_Perturbation_Constant	1015
CPXPARAM_Simplex_Perturbation_Indicator	1027
CPXPARAM_Simplex_PGradient	1029
CPXPARAM_Simplex_Pricing	1010
CPXPARAM_Simplex_Refactor	1031
CPXPARAM_Simplex_Tolerances_Feasibility	1016
CPXPARAM_Simplex_Tolerances_Markowitz	1013
CPXPARAM_Simplex_Tolerances_Optimality	1014
CPXPARAM_SolutionTarget	1131
CPXPARAM_Threads	1067
CPXPARAM_TimeLimit	1039

CPXPARAM_Tune_DetTimeLimit	1139
CPXPARAM_Tune_Display	1113
CPXPARAM_Tune_Measure	1110
CPXPARAM_Tune_Repeat	1111
CPXPARAM_Tune_TimeLimit	1112
CPXPARAM_WorkDir	1064
CPXPARAM_WorkMem	1065

new parameter names introduced in IBM ILOG CPLEX version 12.8.0

Callable library miscellaneous routines

CPXPARAM_Record	1162
CPXPARAM_MIP_Strategy_SubMIPScale	2207
CPXPARAM_MIP_Strategy_SubMIPStartAlg	2205
CPXPARAM_MIP_Strategy_SubMIPSubAlg	2206
CPXPARAM_ParamDisplay	1163
CPX_PARAM_PARAMDISPLAY	1163
CPXMI_SAMECOEFF_ROW	1049
CPXMI_SAMECOEFF_COL	1050
CPXMI_SAMECOEFF_IND	1051
CPXMI_SAMECOEFF_QLIN	1052
CPXMI_SAMECOEFF_QUAD	1053
CPXMI_SAMECOEFF_LAZY	1054
CPXMI_SAMECOEFF_UCUT	1055
CPXMI_SAMECOEFF_RHS	1056
CPXMI_SAMECOEFF_OBJ	1057
CPX_CALLBACKCONTEXT_CANDIDATE	0x0020
CPX_CALLBACKCONTEXT_GLOBAL_PROGRESS	0x0010
CPX_CALLBACKCONTEXT_LOCAL_PROGRESS	0x0008
CPX_CALLBACKCONTEXT_RELAXATION	0x0040
CPX_CALLBACKCONTEXT_THREAD_DOWN	0x0004
CPX_CALLBACKCONTEXT_THREAD_UP	0x0002

new parameter names introduced in IBM ILOG CPLEX version 12.9.0

Callable library miscellaneous routines

CPXPARAM_Read_WarningLimit	1157
CPXPARAM_MultiObjective_Display	1600
CPXPARAM_Preprocessing_Folding	1164

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[return_codeCPLEX](#), [status_codeCPLEX](#), [getParmValCPLEX](#)

cplexError-class *Class "cplexError"*

Description

Objects of class `cpxerr` are returned by various functions of **cplexAPI**, in order to distinguish a status (error) code from a successful result.

Objects from the Class

Objects can be created by calls of the form `cplexError(err)`, with `err` being an error code of IBM ILOG CPLEX.

Slots

`errnum`: Object of class "integer" containing the error code.

Methods

err signature(object = "cplexError"): Prints an error message string corresponding to the error code.

errmsg signature(object = "cplexError"): Returns an error message string corresponding to the error code.

errnum signature(object = "cplexError"): Gets the error code.

errnum<- signature(object = "cplexError"): Sets the error code.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

cplexPtr-class	Class "cplexPtr"
----------------	------------------

Description

Structure of the class "cplexPtr". Objects of that class are used to hold pointers to C structures used by IBM ILOG CPLEX.

Objects from the Class

Objects can be created by calls of the form
 env <-openEnvCPLEX() and/or
 prob <-initProbCPLEX(env).

Slots

cplexPtrType: Object of class "character" giving the pointer type.
 cplexPointer: Object of class "externalptr" containig the pointer to a C structure.

Methods

isCPLEXchanPointer signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a pointer to a CPLEX channel, otherwise FALSE.

isCPLEXenvPointer signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a pointer to a CPLEX environment, otherwise FALSE.

isCPLEXfilePointer signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a pointer to a CPLEX file, otherwise FALSE.

isCPLEXprobPointer signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a pointer to a CPLEX problem object, otherwise FALSE.

isCPLEXtermPointer signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a pointer to a CPLEX termination signal, otherwise FALSE.

isNULLpointerCPLEX signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a NULL pointer, otherwise FALSE.

cplexPointer signature(object = "cplexPtr"): gets the cplexPointer slot.

summary signature(object = "cplexPtr"): prints a summary of the problem object to the command line. If a solution is available, it prints also information retrieved by [solutionCPLEX](#) and [solnInfoCPLEX](#). If no solution is available, it prints the corresponding error message. The method returns invisibly NULL. The CPLEX environment pointer is needed as second argument env to summary.

cplexPtrType signature(object = "cplexPtr"): gets the cplexPtrType slot.

cplexPtrType<- signature(object = "cplexPtr"): sets the cplexPtrType slot.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[openEnvCPLEX](#) and [initProbCPLEX](#)

delColsCPLEX

Delete all Columns in a Specified Range

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delColsCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	Integer value, numeric index of the first column to be deleted.
end	Integer value, numeric index of the last column to be deleted.

Details

Interface to the C function delCols which calls the CPLEX function CPXdelcols.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

delFpDestCPLEX	<i>Remove a File from the List of Message Destinations for a Channel</i>
----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelFpdest. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXdelFpdest has been removed.

Usage

```
delFpDestCPLEX(env, newch, cpfile)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
newch	A pointer to the channel for which destinations are to be deleted as returned by CPXaddchannel .
cpfile	Pointer to an IBM ILOG CPLEX file as returned by openFileCPLEX .

Details

Interface to the C function delFpDest which calls the CPLEX function CPXdelFpdest.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[addFpDestCPLEX](#)

delIndConstrsCPLEX *Delete a Range of Indicator Constraints*

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelindconstrs. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delIndConstrsCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer that specifies the numeric index of the first indicator constraint to be deleted.
end	An integer that specifies the numeric index of the last indicator constraint to be deleted.

Details

Interface to the C function `delIndConstrs` which calls the CPLEX function `CPXdelindconstrs`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

delMIPstartsCPLEX *Delete a Range MIP Starts*

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelmipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delMIPstartsCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the numeric index of the first MIP start to be deleted.
end	An integer specifying the numeric index of the last MIP start to be deleted.

Details

Interface to the C function delMIPstarts which calls the CPLEX function CPXdelmipstarts.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

delNamesCPLEX	<i>Remove all Names Assigned to Rows and Columns</i>
---------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelnames. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delNamesCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function delNames which calls the CPLEX function CPXdelnames.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

delProbCPLEX	<i>Remove Specified CPLEX Problem Object</i>
--------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXfreeprob. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delProbCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function delProb which calls the CPLEX function CPXfreeprob.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[initProbCPLEX](#)

delQConstrsCPLEX *Delete a Range of Quadratic Constraints*

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelqconstrs. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delQConstrsCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer that specifies the numeric index of the first quadratic constraint to be deleted.
end	An integer that specifies the numeric index of the last quadratic constraint to be deleted.

Details

Interface to the C function `delQConstrs` which calls the CPLEX function `CPXdelqconstrs`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

delRowsCPLEX	<i>Delete a Range of Rows</i>
--------------	-------------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delRowsCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	Integer value, numeric index of the first row to be deleted.
end	Integer value, numeric index of the last row to be deleted.

Details

Interface to the C function delRows which calls the CPLEX function CPXdelrows.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

delSetColsCPLEX *Delete a Set of Columns*

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelsetcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delSetColsCPLEX(env, lp, delstat)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
delstat	An array specifying the columns to be deleted.

Details

Interface to the C function `delSetCols` which calls the CPLEX function `CPXdelsetcols`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

delSetRowsCPLEX	<i>Delete a Set of Rows</i>
-----------------	-----------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelsetrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delSetRowsCPLEX(env, lp, delstat)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
delstat	An array specifying the rows to be deleted.

Details

Interface to the C function `delSetRows` which calls the CPLEX function `CPXdelsetrows`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

delTerminateCPLEX *Terminate CPLEX gracefully*

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetterminate. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delTerminateCPLEX(env, tsig)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
tsig	Pointer to termination signal as returned by setTerminateCPLEX .

Details

Interface to the C function setTerminate which calls the CPLEX function CPXsetterminate.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[setTerminateCPLEX](#), [printTerminateCPLEX](#), [chgTerminateCPLEX](#)

`disconnectChannelCPLEX`*Flush all Message Destinations Associated with a Channel*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXdisconnectchannel`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
disconnectChannelCPLEX(env, newch)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>newch</code>	A pointer to the channel containing the message destinations as returned by <code>CPXaddchannel</code> .

Details

Interface to the C function `disconnectChannel` which calls the CPLEX function `CPXdisconnectchannel`.

Value

NULL

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[flushChannelCPLEX](#), [flushStdChannelsCPLEX](#), [getChannelsCPLEX](#)

`dualoptCPLEX`*Find a Problem Solution Using the Dual Simplex Algorithm*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXdualopt`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
dualoptCPLEX(env, lp)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `dualopt` which calls the CPLEX function `CPXdualopt`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[solnInfoCPLEX](#), [getStatCPLEX](#), [solutionCPLEX](#)

dualWriteCPLEX	<i>Write a Dual Formulation of the Current CPLEX Problem Object</i>
----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function `CPXdualwrite`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
dualWriteCPLEX(env, lp, fname)
```

Arguments

env	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
fname	Single character value giving the filename to write to.

Details

Interface to the C function `dualWrite` which calls the CPLEX function `CPXdualwrite`.

Value

Zero if successful, otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`feasOptCPLEX`*Compute a Minimum-Cost Relaxation*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXfeasopt`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
feasOptCPLEX(env, lp, rhs = FALSE, rng = FALSE, lb = FALSE, ub = FALSE)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>rhs</code>	If set to <code>FALSE</code> no right hand side value is allowed to be relaxed.
<code>rng</code>	If set to <code>FALSE</code> no range values are allowed to be relaxed.
<code>lb</code>	If set to <code>FALSE</code> no lower bound of any variable is allowed to be relaxed.
<code>ub</code>	If set to <code>FALSE</code> no upper bound of any variable is allowed to be relaxed.

Details

Interface to the C function `feasOpt` which calls the CPLEX function `CPXfeasopt`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[solutionCPLEX](#), [getRowInfeasCPLEX](#), [getColInfeasCPLEX](#), [solnInfoCPLEX](#), [getStatCPLEX](#)

fileputCPLEX	<i>Write to File</i>
--------------	----------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXfputs. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXfputs has been removed.

Usage

```
fileputCPLEX(cpfile, stuff = "")
```

Arguments

cpfile	A pointer to a file as returned by openFileCPLEX .
stuff	A character string to be written to the file.

Details

Interface to the C function fileput which calls the CPLEX function CPXfputs.

Value

A nonnegative value if successful, otherwise EOF.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[closeFileCPLEX](#), [openFileCPLEX](#)

flushChannelCPLEX *Flush All Message Destinations Associated With a Channel*

Description

Low level interface function to the IBM ILOG CPLEX function CPXflushchannel. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
flushChannelCPLEX(env, newch)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
newch	Pointer to a channel object as returned by addChannelCPLEX .

Details

Interface to the C function flushChannel which calls the CPLEX function CPXflushchannel.

Value

NULL

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[disconnectChannelCPLEX](#), [flushStdChannelsCPLEX](#), [getChannelsCPLEX](#)

flushStdChannelsCplex *Flushes the Output Buffers of the Four Standard Channels*

Description

Low level interface function to the IBM ILOG CPLEX function CPXflushstdchannels. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
flushStdChannelsCplex(env)
```

Arguments

env An object of class "cplexPtr" as returned by [openEnvCplex](#). This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function flushStdChannels which calls the CPLEX function CPXflushstdchannels.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[disconnectChannelCplex](#), [flushChannelCplex](#), [getChannelsCplex](#)

freePresolveCPLEX *Free Presolved Problem From the LP Problem Object*

Description

Low level interface function to the IBM ILOG CPLEX function CPXfreepresolve. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
freePresolveCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function freePresolve which calls the CPLEX function CPXfreepresolve.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getBaseCPLEX *Access Basis Resident in a CPLEX Problem Object.*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetbase. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getBaseCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getBase which calls the CPLEX function CPXgetbase.

Value

If successful a list is returned:

cstat	basis status of the columns in the CPLEX problem object
rstat	basis status of the artificial, slack, or surplus variable associated with each row in the constraint matrix

otherwise an instance of class "[cplexError](#)".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getBestObjValCPLEX *Access the Currently Best Known Bound of all the Remaining Open Nodes in a Branch-And-Cut Tree*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetbestobjval. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getBestObjValCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getBestObjVal` which calls the CPLEX function `CPXgetbestobjval`.

Value

Objective value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getChannelsCPLEX *Obtain Pointers to the Four Default Channels*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetchannels. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getChannelsCPLEX(env, ptrtype = "cplex_chan")
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
ptrtype	A name for the pointer object.

Details

Interface to the C function `getChannels` which calls the CPLEX function `CPXgetchannels`.

Value

If successful a list is returned:

cpxresults	address of the channel corresponding to cpxresults
cpxwarning	address of the channel corresponding to cpxwarning
cpxerror	address of the channel corresponding to cpxerror
cpxlog	address of the channel corresponding to cpxlog

otherwise an instance of class "cplexError". Each list element is an object of class "cplexPtr".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[disconnectChannelCPLEX](#), [flushChannelCPLEX](#), [flushStdChannelsCPLEX](#)

getChgParmCPLEX	<i>Get Parameter Numbers for Parameters Which are Not Set at Their Default Values</i>
-----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetchgparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getChgParmCPLEX(env)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
-----	---

Details

Interface to the C function getChgParm which calls the CPLEX function CPXgetchgparam.

Value

A vector containing integer values (unique parameter identifiers) for parameters which are not set at their default values, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getCoefCPLEX	<i>Access a Single Constraint Matrix Coefficient</i>
--------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcoef. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getCoefCPLEX(env, lp, i, j)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
i	An integer specifying the numeric index of the row.
j	An integer specifying the numeric index of the column.

Details

Interface to the C function `getCoef` which calls the CPLEX function `CPXgetcoef`.

Value

Matrix coefficient value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getColIndexCPLEX *Search for the Index Number of the Specified Column*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcolindex. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getColIndexCPLEX(env, lp, cname)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
cname	A column name to search for.

Details

Interface to the C function `getColIndex` which calls the CPLEX function `CPXgetcolindex`.

Value

Column number if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getColInfeasCPLEX *Compute Infeasibility of a Given Solution for a Range of Variables*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcolinfeas. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getColInfeasCPLEX(env, lp, begin, end, sol = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of variables whose infeasibility is to be returned.
end	An integer specifying the end of the range of variables whose infeasibility is to be returned.
sol	The solution whose infeasibility is to be computed.

Details

Interface to the C function `getColInfeas` which calls the CPLEX function `CPXgetcolinfeas`.

Value

infeasibility values if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

getColNameCPLEX	<i>Access a Range of Column Names</i>
-----------------	---------------------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcolname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getColNameCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of column names to be returned.
end	An integer specifying the end of the range of column names to be returned.

Details

Interface to the C function `getColName` which calls the CPLEX function `CPXgetcolname`.

Value

Column names if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getColsCPLEX	<i>Accesses a Range of Columns of the Constraint Matrix</i>
--------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getColsCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of columns to be returned.
end	An integer specifying the end of the range of columns to be returned.

Details

Interface to the C function `getCols` which calls the CPLEX function `CPXgetcols`.

Value

If successful a list is returned:

matbeg	Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind	Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval	Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`getColTypeCPLEX`*Access Types for a Range of Variables*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXgetcctype`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getColTypeCPLEX(env, lp, begin, end)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>begin</code>	An integer specifying the beginning of the range of the types to be returned.
<code>end</code>	An integer specifying the end of the range of the types to be returned.

Details

Interface to the C function `getColType` which calls the CPLEX function `CPXgetcctype`.

Value

Column types if successful, otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getConflictCPLEX *Return Linear Constraints and Variables Belonging to a Conflict*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetconflict. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getConflictCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getConflict` which calls the CPLEX function `CPXgetconflict`.

Value

If successful a list is returned:

confstat	status of the conflict
confnumrows	number of rows in the conflict
rowind	indices of the constraints that participate in the conflict
rowbdstat	conflict status of the rows
confnumcols	number of columns in the conflict
colind	indices of the variables that participate in the conflict
colbdstat	conflict status of the columns

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getConflictExtCPLEX *Get Conflict Status Codes*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetconflicttext. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getConflictExtCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	The index of the first group.
end	The index of the last group.

Details

Interface to the C function getConflictExt which calls the CPLEX function CPXgetconflicttext.

Value

Specified values denoting the conflict status if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getCutoffCPLEX	<i>Access MIP Cutoff Value Being Used During Mixed Integer Optimization.</i>
----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcutoff. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getCutoffCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getCutoff` which calls the CPLEX function `CPXgetcutoff`.

Value

Value of the cutoff if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getDb1ParmCPLEX *Obtain the Current Value of a CPLEX Parameter of Type Double*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetdblparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getDb1ParmCPLEX(env, parm)
```

Arguments

env	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.

Details

Interface to the C function `getDb1Parm` which calls the CPLEX function `CPXgetdblparam`.

Value

Parameter value if successful, otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getDblQualCPLEX	<i>Access Double-Valued Information About the Quality of the Current Solution of a Problem</i>
-----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetdblquality. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getDblQualCPLEX(env, lp, w)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
w	An Integer specifying the quality value to be retrieved.

Details

Interface to the C function `getDblQual` which calls the CPLEX function `CPXgetdblquality`.

Value

Requested quality value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getDbsCntCPLEX	<i>Access the Number of Dual Super-Basic Variables in the Current Solution</i>
----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetdsbcnt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getDbsCntCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getDbsCnt` which calls the CPLEX function `CPXgetdsbcnt`.

Value

Number of dual super-basic variables if a solution exists, otherwise zero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

getDjCPLEX	<i>Accesses Reduced Costs for a Range of Variables of a Linear or Quadratic Program</i>
------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetdj. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getDjCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of reduced-cost values to be returned.
end	An integer specifying the end of the range of reduced-costs values to be returned.

Details

Interface to the C function `getDj` which calls the CPLEX function `CPXgetdj`.

Value

Reduced costs if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getErrorStrCPLEX *Return an Error Message String Corresponding to an Error Code*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgeterrorstring. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getErrorStrCPLEX(err, env = NULL)
```

Arguments

err	The error code to be translated.
env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function getErrorStr which calls the CPLEX function CPXgeterrorstring.

Value

A single character value containing the error message string.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[getStatStrCPLEX](#)

getGradCPLEX	<i>Project the Impact of Making Changes to Optimal Variable Values or Objective Function Coefficients</i>
--------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetgrad. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getGradCPLEX(env, lp, j)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
j	An integer specifying the index of the column of interest.

Details

Interface to the C function getGrad which calls the CPLEX function CPXgetgrad.

Value

If successful a list is returned:

head	listing of the indices of the basic variables in the order in which they appear in the basis.
y	coefficients of the j-th column relative to the current basis.

otherwise an instance of class "[cplexError](#)".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getIndConstrCPLEX	<i>Access a Specified Indicator Constraint on the Variables of a CPLEX Problem Object.</i>
-------------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetindconstr. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getIndConstrCPLEX(env, lp, which)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
which	An integer specifying which indicator constraint to return.

Details

Interface to the C function getIndConstr which calls the CPLEX function CPXgetindconstr.

Value

If successful a list is returned:

indvar	Index of the binary indicator variable. Consult the IBM ILOG CPLEX documentation for more detailed information.
complemented	Boolean value that specifies whether the indicator variable is complemented. Consult the IBM ILOG CPLEX documentation for more detailed information.
rhs	Righthand side value of the linear portion of the indicator constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.
sense	Sense of the linear portion of the constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.
linind	Variable indices of the entries of linval. Consult the IBM ILOG CPLEX documentation for more detailed information.
linval	Coefficients of the linear portion of the specified indicator constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "[cplexError](#)".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getInfoDb1ParmCPLEX *Obtain Default, Minimum and Maximum Values of a Parameter of Type Double*

Description

Low level interface function to the IBM ILOG CPLEX function CPXinfodblparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getInfoDb1ParmCPLEX(env, parm)
```

Arguments

env An object of class "cplexPtr" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.

parm Constant or reference number of the desired parameter.

Details

Interface to the C function getInfoDb1Parm which calls the CPLEX function CPXinfodblparam.

Value

If successful a list is returned:

defvalue default value

minvalue minimum value

maxvalue maximum value

otherwise an instance of class "[cplexError](#)".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getInfoIntParmCPLEX	<i>Obtain Default, Minimum and Maximum Values of a Parameter of Type CPXINT</i>
---------------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXinfointparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getInfoIntParmCPLEX(env, parm)
```

Arguments

env	An object of class " cplexPtr " as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.

Details

Interface to the C function `getInfoIntParm` which calls the CPLEX function `CPXinfointparam`.

Value

If successful a list is returned:

defvalue	default value
minvalue	minimum value
maxvalue	maximum value

otherwise an instance of class "[cplexError](#)".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also[cplexConstants](#)

getInfoLongParmCPLEX *Obtain Default, Minimum and Maximum Values of a Parameter of Type CPXLONG*

Description

Low level interface function to the IBM ILOG CPLEX function CPXinfolongparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getInfoLongParmCPLEX(env, parm)
```

Arguments

env	An object of class " cplexPtr " as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.

Details

Interface to the C function getInfoLongParm which calls the CPLEX function CPXinfolongparam.

Value

If successful a list is returned:

defvalue	default value
minvalue	minimum value
maxvalue	maximum value

otherwise an instance of class "[cplexError](#)".

Note

In order to get a 64 bit integer value from CPXinfolongparam, datatype numeric is used. All return values will be numeric.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[getInfoIntParmCPLEX](#), [cplexConstants](#)

getInfoStrParmCPLEX *Obtain Default Value of a String Parameter*

Description

Low level interface function to the IBM ILOG CPLEX function CPXinfostrparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getInfoStrParmCPLEX(env, parm)
```

Arguments

env	An object of class " cplexPtr " as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.

Details

Interface to the C function `getInfoStrParm` which calls the CPLEX function `CPXinfostrparam`.

Value

A single character value.

Author(s)

Gabriel Gelius-Dietrich <geljudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getIntParmCPLEX	<i>Obtain the Current Value of a CPLEX Parameter of Type CPXINT</i>
-----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetintparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getIntParmCPLEX(env, parm)
```

Arguments

env	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.

Details

Interface to the C function `getIntParm` which calls the CPLEX function `CPXgetintparam`.

Value

Parameter value if successful, otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getIntQualCPLEX	<i>Access Integer-Valued Information About the Quality of the Current Solution of a Problem</i>
-----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetintquality. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getIntQualCPLEX(env, lp, w)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
w	An Integer specifying the quality value to be retrieved.

Details

Interface to the C function `getIntQual` which calls the CPLEX function `CPXgetintquality`.

Value

Requested quality value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getItCntCPLEX	<i>Access the Total Number of Simplex Iterations to Solve an LP Problem</i>
---------------	---

Description

Low level interface function to the IBM ILOG CPLEX function `CPXgetitcnt`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getItCntCPLEX(env, lp)
```

Arguments

env	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getItCnt` which calls the CPLEX function `CPXgetitcnt`.

Value

Total iteration count if solution exists, otherwise zero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getLogFileCPLEX	<i>Access log file to Which Messages are Written</i>
-----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetlogfile. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXgetlogfile has been removed.

Usage

```
getLogFileCPLEX(env, ptrtype = "cplex_file")
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
ptrtype	A name for the pointer object.

Details

Interface to the C function getLogFile which calls the CPLEX function CPXgetlogfile.

Value

If successful, a pointer to the CPLEX file is returned (an instance of class "cplexPtr"), otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[setLogFileCPLEX](#)

getLogFileNameCPLEX *Get the name of the current logfile*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetlogfile. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality for IBM ILOG CPLEX < 12.8.0, where CPXgetlogfile was not included.

Usage

```
getLogFileNameCPLEX(env)
```

Arguments

env An object of class "cplexPtr" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function getLogFileName which calls the CPLEX function CPXgetlogfile.

Value

Zero if successful, otherwise nonzero.

Author(s)

Mayo Roettger <mayo.roettger@hhu.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getLongParmCPLEX *Obtain Current Value of a Parameter of Type CPXLONG*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetlongparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getLongParmCPLEX(env, parm)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.

Details

Interface to the C function getLongParm which calls the CPLEX function CPXgetlongparam.

Value

Parameter value if successful, otherwise an instance of class "cplexError".

Note

In order to get a 64 bit integer value from CPXgetlongparam, datatype numeric is used. The return value will be numeric.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[getIntParmCPLEX](#), [cplexConstants](#)

getLowBndsIdsCPLEX *Retrieve Lower Bounds on Variables*

Description

The function retrieves the lower bounds on specified variables.

Usage

```
getLowBndsIdsCPLEX(env, lp, ind)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
ind	Column indices of variables (remember: first index is 0).

Value

A numeric vector containing the lower bounds on the specified variables. If not successful an instance of class "cplexError" is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[getLowerBndsCPLEX](#)

getLowerBndsCPLEX *Access a Range of Lower Bounds on Variables*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetlb. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getLowerBndsCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	Beginning of the range of lower bounds to be returned.
end	End of the range of lower bounds to be returned.

Details

Interface to the C function `getLowerBnds` which calls the CPLEX function `CPXgetlb`.

Value

A numeric vector containing the lower bounds on the specified variables. If not successful an instance of class "cplexError" is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getMethodCPLEX	<i>Obtain Solution Algorithm</i>
----------------	----------------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetmethod. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getMethodCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getMethod` which calls the CPLEX function `CPXgetmethod`.

Value

A single integer value specifying the solution algorithm.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#) section "LP/QP solution algorithms".

getMIPrelGapCPLEX *Access Relative Objective Gap for a MIP Optimization*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetmiprelgap. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getMIPrelGapCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getMIPrelGap which calls the CPLEX function CPXgetmiprelgap.

Value

Relative Objective Gap if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[getObjValCPLEX](#), [getBestObjValCPLEX](#)

getMIPstartIndexCPLEX *Search for the Index Number of the Specified MIP Start*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetmipstartindex. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getMIPstartIndexCPLEX(env, lp, iname)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
iname	A MIP start name to search for.

Details

Interface to the C function `getMIPstartIndex` which calls the CPLEX function `CPXgetmipstartindex`.

Value

Index number of the specified MIP start if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getMIPstartNameCPLEX *Access a Range of Names of MIP Starts*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetmipstartname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getMIPstartNameCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of MIP starts to be returned.
end	An integer specifying the end of the range of MIP starts to be returned.

Details

Interface to the C function `getMIPstartName` which calls the CPLEX function `CPXgetmipstartname`.

Value

Names of the MIP starts if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getMIPstartsCPLEX *Access a Range of MIP Starts of a CPLEX Problem Object*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetmipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getMIPstartsCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of MIP starts to be returned.
end	An integer specifying the end of the range of MIP starts to be returned.

Details

Interface to the C function getMIPstarts which calls the CPLEX function CPXgetmipstarts.

Value

If successful a list is returned:

beg	Array specifying where each of the requested MIP starts begins in the arrays varindices and values. Consult the IBM ILOG CPLEX documentation for more detailed information.
varindices	Array containing the numeric indices of the columns corresponding to the variables which are assigned starting values. Consult the IBM ILOG CPLEX documentation for more detailed information.
values	Array containing the values of the MIP starts. Consult the IBM ILOG CPLEX documentation for more detailed information.
effortlevel	Array containing the effort level for each MIP start requested. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "[cplexError](#)".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getNumColsCPLEX

Access the Number of Columns in the Constraint Matrix

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnumcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getNumColsCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getNumCols` which calls the CPLEX function `CPXgetnumcols`.

Value

If successful the number of variables is returned. If env or lp do not exist, zero is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getNumMIPstartsCPLEX *Access the Number of MIP Starts in the CPLEX Problem Object*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnummipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getNumMIPstartsCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getNumMIPstarts` which calls the CPLEX function `CPXgetnummipstarts`.

Value

If successful the number of MIP starts is returned. If env or lp do not exist, zero is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getNumNnzCplex	<i>Access the Number of Nonzero Elements in the Constraint Matrix</i>
----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnumnz. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getNumNnzCplex(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCplex . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCplex . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getNumNnz which calls the CPLEX function CPXgetnumnz.

Value

Zero if the problem object or environment does not exist, otherwise the number of nonzero elements.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getNumQConstrsCPLEX *Return the Number of quadratic constraints.*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnumqconstrs. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getNumQConstrsCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getNumQConstrs which calls the CPLEX function CPXgetnumqconstrs.

Value

If successful the number of quadratic constraints is returned. If env or lp do not exist, zero is returned.

Author(s)

Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

getNumQPnzCPLEX	<i>Return the Number of Nonzeros in the Q Matrix</i>
-----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnumqpnz. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getNumQPnzCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getNumQPnz` which calls the CPLEX function `CPXgetnumqpnz`.

Value

If successful the number of nonzeros in the Q matrix is returned. If env or lp do not exist, zero is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getNumQuadCPLEX	<i>Return the Number of Variables That Have Quadratic Objective Coefficients</i>
-----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnumquad. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getNumQuadCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getNumQuad` which calls the CPLEX function `CPXgetnumquad`.

Value

If successful the number of variables that have quadratic objective coefficients is returned. If `env` or `lp` do not exist, zero is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getNumRowsCPLEX *Access the Number of Rows in the Constraint Matrix*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnumcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getNumRowsCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getNumRows` which calls the CPLEX function `CPXgetnumrows`.

Value

If successful the number of rows is returned. If env or lp do not exist, zero is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getObjCPLEX	<i>Access a Range of Objective Function Coefficients of a CPLEX Problem Object</i>
-------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetobj. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getObjCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of objective function coefficients to be returned
end	An integer specifying the end of the range of objective function coefficients to be returned.

Details

Interface to the C function `getObj` which calls the CPLEX function `CPXgetobj`.

Value

Specified objective coefficients if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getObjDirCPLEX *Access the Direction of Optimization*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetobjsen. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getObjDirCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getObjDir` which calls the CPLEX function `CPXgetobjsen`.

Value

Zero if the problem object or environment does not exist, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getObjNameCPLEX	<i>Access the Name of the Objective Row of a CPLEX Problem Object</i>
-----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetobjname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getObjNameCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getObjName which calls the CPLEX function CPXgetobjname.

Value

Name of the objective row if successful, otherwise an instance of class "[cplexError](#)".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getObjOffsetCPLEX	<i>Objective Offset Between the Original Problem and the Presolved Problem.</i>
-------------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetobjoffset. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getObjOffsetCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getObjOffset` which calls the CPLEX function `CPXgetobjoffset`.

Value

Objective offset value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

getObjValCPLEX	<i>Access Solution Objective Value</i>
----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetobjval. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getObjValCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getObjVal` which calls the CPLEX function `CPXgetobjval`.

Value

Objective value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getOrderCPLEX *Access MIP Priority Order Information*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetorder. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getOrderCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getOrder` which calls the CPLEX function `CPXgetorder`.

Value

If successful a list is returned:

indices	indices of the variables in the order
priority	priority values
direction	preferred branching directions

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getParmHierNameCPLEX *Obtain the hierarchy name string of a CPLEX parameter*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetparamhiername. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality for IBM ILOG CPLEX < 12.9.0 on, where CPXgetparamhiername was not included.

Usage

```
getParmHierNameCPLEX(env, whichparam)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
whichparam	An integer specifying the symbolic constant (or reference number) of the desired parameter.

Details

Interface to the C function getParmHierName which calls the CPLEX function CPXgetparamhiername.

Value

A single character value.

Author(s)

Mayo Roettger <mayo.roettger@hhu.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getParmNameCPLEX *Obtain the Name of a CPLEX Parameter, Given the Symbolic Constant*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetparamname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getParmNameCPLEX(env, wparm)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
wparm	Constant or reference number of the desired parameter.

Details

Interface to the C function getParmName which calls the CPLEX function CPXgetparamname.

Value

A single character value.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getParmNumCPLEX	<i>Obtain the Reference Number of a CPLEX Parameter</i>
-----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetparamnum. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getParmNumCPLEX(env, nparam)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
nparam	A single character value containing the name of the parameter.

Details

Interface to the C function getParmNum which calls the CPLEX function CPXgetparamnum.

Value

A single integer value.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[cplexConstants](#)

getParmTypeCPLEX *Obtain the Type of a CPLEX Parameter*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetparamtype. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getParmTypeCPLEX(env, parm)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.

Details

Interface to the C function getParmType which calls the CPLEX function CPXgetparamtype.

Value

A single integer value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[cplexConstants](#)

getParmValCPLEX	<i>Values and Names of Parameters Having Non-Default Values</i>
-----------------	---

Description

The function `getParmValCPLEX` retrieves the names and actual values of all IBM ILOG CPLEX parameters, which do not have their default values.

Usage

```
getParmValCPLEX(env)
```

Arguments

env	An object of class " <code>cpLexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
-----	---

Value

Either a list containing all non-default parameters and their values or NULL.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cpLexConstants](#) and [getChgParmCPLEX](#)

getPhase1CntCPLEX	<i>Access Number of Phase I Iterations</i>
-------------------	--

Description

Low level interface function to the IBM ILOG CPLEX function `CPXgetphase1cnt`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getPhase1CntCPLEX(env, lp)
```

Arguments

- env An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getPhase1Cnt` which calls the CPLEX function `CPXgetphase1cnt`.

Value

Zero if no solution exists, otherwise Phase I iteration count.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getPiCplex

Access Dual Values for a Range of Constraints

Description

Low level interface function to the IBM ILOG CPLEX function `CPXgetpi`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getPiCplex(env, lp, begin, end)
```

Arguments

- env An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- begin An integer specifying the beginning of the range of dual values to be returned.
- end An integer specifying the end of the range of dual values to be returned.

Details

Interface to the C function `getPi` which calls the CPLEX function `CPXgetpi`.

Value

Values of the dual variables if successful, otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getPreStatCPLEX	<i>Access Presolve Status Information for Columns and Rows</i>
-----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function `CPXgetprestat`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getPreStatCPLEX(env, lp)
```

Arguments

`env` An object of class "`cplexPtr`" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

`lp` An object of class "`cplexPtr`" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getPreStat` which calls the CPLEX function `CPXgetprestat`.

Value

If successful a list is returned:

<code>prestat</code>	status of the presolved problem
<code>pcstat</code>	presolve status values of the columns
<code>prstat</code>	presolve status values of the rows
<code>ocstat</code>	presolve status values of the columns of the presolved problem
<code>orstat</code>	presolve status values of the rows of the presolved problem

otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getProbNameCPLEX	<i>Access Problem Name</i>
------------------	----------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetprobname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getProbNameCPLEX(env, lp)
```

Arguments

env An object of class "cplexPtr" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by [initProbCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getProbName which calls the CPLEX function CPXgetprobname.

Value

Name of the problem if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getProbTypeCPLEX *Access Problem Type*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetprobtype. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getProbTypeCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getProbType which calls the CPLEX function CPXgetprobtype.

Value

A single integer value specifying the problem type.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[chgProbTypeCPLEX](#), [cplexConstants](#) section "Problem Types".

getProbVarCPLEX *Access the Solution Values for a Range of Problem Variables*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetx. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getProbVarCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of variable values to be returned.
end	An integer specifying the end of the range of variable values to be returned.

Details

Interface to the C function `getProbVar` which calls the CPLEX function `CPXgetx`.

Value

Values of the primal variables if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getQConstrCPLEX	<i>Access a Specified Quadratic Constraint on the Variables of a CPLEX Problem Object.</i>
-----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetqconstr. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getQConstrCPLEX(env, lp, which)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
which	An integer specifying which quadratic constraint to return.

Details

Interface to the C function getQConstr which calls the CPLEX function CPXgetqconstr.

Value

If successful a list is returned:

rhs	Righthand-side value of the quadratic constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.
sense	Character specifying the sense of the constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.
linind	Variable indices of the entries of linval. Consult the IBM ILOG CPLEX documentation for more detailed information.
linval	Linear coefficients of the specified constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.
quadrow	Variable indices of the entries of quadval. Consult the IBM ILOG CPLEX documentation for more detailed information.
quadcol	Variable indices of the entries of quadval. Consult the IBM ILOG CPLEX documentation for more detailed information.
quadval	Quadratic coefficients of the specified constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getQPcoefCPLEX

Access the Quadratic Coefficient in the Matrix Q

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetqpcoef. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getQPcoefCPLEX(env, lp, i, j)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
i	The row number in Q.
j	The row column in Q.

Details

Interface to the C function `getQPcoef` which calls the CPLEX function `CPXgetqpcoef`.

Value

Specified quadratic coefficient in the matrix Q if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getQuadCPLEX	<i>Access a Range of Columns of the Matrix Q of a Model With a Quadratic Objective Function</i>
--------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetquad. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getQuadCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of columns to be returned.
end	An integer specifying the end of the range of columns to be returned.

Details

Interface to the C function `getQuad` which calls the CPLEX function `CPXgetquad`.

Value

If successful a list is returned:

qmatbeg	Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.
qmatind	Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.
qmatval	Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`getRedLpCPLEX`*Get a Pointer for the Presolved Problem*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXgetredlp`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getRedLpCPLEX(env, lp, ptrtype = "cplex_prob")
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>ptrtype</code>	A name for the pointer object.

Details

Interface to the C function `getRedLp` which calls the CPLEX function `CPXgetredlp`.

Value

Pointer for the presolved problem if successful (an instance of class "`cplexPtr`"), otherwise an instance of class "`cplexError`" or `NULL`.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`getRhsCPLEX`*Access Righthand Side Coefficients for a Range of Constraints*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetrhs. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getRhsCPLEX(env, lp, begin, end)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>begin</code>	An integer specifying the beginning of the range of righthand side terms to be returned.
<code>end</code>	An integer specifying the end of the range of righthand side terms to be returned.

Details

Interface to the C function `getRhs` which calls the CPLEX function `CPXgetrhs`.

Value

Specified righthand side coefficients if successful, otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getRngValCPLEX *Accesses Righthand Side Coefficients Range Coefficients*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetrngval. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getRngValCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the set of rows for which RHS range coefficients are returned.
end	An integer specifying the end of the set of rows for which RHS range coefficients are returned.

Details

Interface to the C function `getRngVal` which calls the CPLEX function `CPXgetrngval`.

Value

Specified RHS range coefficients if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getRowIndexCPLEX	<i>Search for the Index Number of a Specified Row</i>
------------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetrowindex. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getRowIndexCPLEX(env, lp, rname)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
rname	A row name to search for.

Details

Interface to the C function `getRowIndex` which calls the CPLEX function `CPXgetrowindex`.

Value

Specified row index if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getRowInfeasCPLEX	<i>Compute Infeasibility of a Given Solution for a Range of Linear Constraints</i>
-------------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetrowinfeas. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getRowInfeasCPLEX(env, lp, begin, end, sol = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of linear constraints whose infeasibility is to be returned.
end	An integer specifying the beginning of the range of linear constraints whose infeasibility is to be returned.
sol	The solution whose infeasibility is to be computed.

Details

Interface to the C function `getRowInfeas` which calls the CPLEX function `CPXgetrowinfeas`.

Value

Infeasibility values if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getRowNameCPLEX	<i>Access a Range of Row Names</i>
-----------------	------------------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetrowname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getRowNameCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of row names to be returned.
end	An integer specifying the end of the range of row names to be returned.

Details

Interface to the C function getRowName which calls the CPLEX function CPXgetrowname.

Value

Specified row names if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getRowsCPLEX	<i>Accesses a Range of Rows of the Constraint Matrix</i>
--------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getRowsCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of rows to be returned.
end	An integer specifying the end of the range of rows to be returned.

Details

Interface to the C function `getRows` which calls the CPLEX function `CPXgetrows`.

Value

If successful a list is returned:

matbeg	Array that specifies the nonzero elements of the rows. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind	Array that specifies the nonzero elements of the rows. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval	Array that specifies the nonzero elements of the rows. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
 Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getSenseCPLEX	<i>Access the Sense for a Range of Constraints in a CPLEX Problem Object.</i>
---------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetsense. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getSenseCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of constraint senses to be returned.
end	An integer specifying the end of the range of constraint senses to be returned.

Details

Interface to the C function getSense which calls the CPLEX function CPXgetsense.

Value

Specified constraint senses if successful, otherwise an instance of class "[cplexError](#)".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getSiftItCntCPLEX *Access Total Number of Sifting Iterations*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetsiftitcnt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getSiftItCntCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getSiftItCnt which calls the CPLEX function CPXgetsiftitcnt.

Value

Zero if no solution exists, otherwise nonzero the total iteration count.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getSiftPhase1CntCPLEX *Access Number of Phase I Sifting Iterations*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetsiftphase1cnt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getSiftPhase1CntCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getSiftPhase1Cnt which calls the CPLEX function CPXgetsiftphase1cnt.

Value

Zero if no solution exists, otherwise nonzero the Phase I iteration count.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getSlackCPLEX *Accesses Slack Values for a Range of Linear Constraints*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetslack. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getSlackCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	An integer specifying the beginning of the range of slack values to be returned.
end	An integer specifying the end of the range of slack values to be returned.

Details

Interface to the C function getSlack which calls the CPLEX function CPXgetslack.

Value

Specified slack or surplus variables if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getStatCPLEX *Access the Solution Status of the Problem*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetstat. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getStatCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getStat which calls the CPLEX function CPXgetstat.

Value

A single integer value giving the solution status.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#) section "Values returned for stat by solution".

getStatStrCPLEX *Return an Status Message String Corresponding to an Status Code*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetstatstring. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getStatStrCPLEX(env, stat)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
stat	The status code to be translated.

Details

Interface to the C function getStatStr which calls the CPLEX function CPXgetstatstring.

Value

A single character value containing the status message string.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[getErrorStrCPLEX](#)

getStrParmCPLEX *Obtain the Current Value of a CPLEX String Parameter*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetstrparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getStrParmCPLEX(env, parm)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.

Details

Interface to the C function getStrParm which calls the CPLEX function CPXgetstrparam.

Value

A single character value.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getSubMethodCPLEX *Accesses Solution Method of the Last Subproblem Optimization*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetsubmethod. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getSubMethodCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getSubMethod which calls the CPLEX function CPXgetsubmethod.

Value

Integer value specifying the solution method.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

getSubStatCPLEX	<i>Access Solution Status of the Last Subproblem Optimization</i>
-----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetsubstat. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getSubStatCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getSubStat which calls the CPLEX function CPXgetsubstat.

Value

Zero if no solution exists, nonzero otherwise.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getTimeCPLEX	<i>Get a Time Stamp</i>
--------------	-------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXfclose. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getTimeCPLEX(env)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
-----	--

Details

Interface to the C function `getTime` which calls the CPLEX function `CPXgetTime`.

Value

If successful a single numeric value, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

`fileputCPLEX`, `openFileCPLEX`

getUppBndsIdsCPLEX *Retrieve Upper Bounds on Variables*

Description

The function retrieves the upper bounds on specified variables.

Usage

```
getUppBndsIdsCPLEX(env, lp, ind)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
ind	Column indices of variables (remember: first index is 0).

Value

A numeric vector containing the upper bounds on the specified variables. If not successful an instance of class "cplexError" is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[getUpperBndsCPLEX](#)

getUpperBndsCPLEX *Access a Range of Upper Bounds on Variables*

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetub. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getUpperBndsCPLEX(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	Beginning of the range of upper bounds to be returned.
end	End of the range of upper bounds to be returned.

Details

Interface to the C function `getUpperBnds` which calls the CPLEX function `CPXgetub`.

Value

A numeric vector containing the lower bounds on the specified variables. If not successful an instance of class "cplexError" is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

getVersionCPLEX *Get Version Number of the CPLEX Library.*

Description

Low level interface function to the IBM ILOG CPLEX function CPXversion. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
getVersionCPLEX(env)
```

Arguments

env An object of class "cplexPtr" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function getVersion which calls the CPLEX function getVersionCPLEX.

Value

Single character string specifying the version of the cplex library or NULL if the environment does not exist.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

hybbaroptCPLEX *Solve the Specified Problem by the CPLEX Barrier Optimizer*

Description

Low level interface function to the IBM ILOG CPLEX function CPXhybbaropt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
hybbaroptCPLEX(env, lp, method)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
method	A single integer value giving the crossover method to be implemented.

Details

Interface to the C function `hybbaropt` which calls the CPLEX function `CPXhybbaropt`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

`solnInfoCPLEX`, `getStatCPLEX`, `solutionCPLEX`, `cplexConstants` section "LP/QP solution algorithms".

hybnetoptCPLEX *Use CPLEX Network Optimizer*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXhybnetopt`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
hybnetoptCPLEX(env, lp, method)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
method	A single integer value giving the type of simplex method to follow the network optimization.

Details

Interface to the C function `hybnetopt` which calls the CPLEX function `CPXhybnetopt`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[solnInfoCPLEX](#), [getStatCPLEX](#), [solutionCPLEX](#), [cplexConstants](#) section "LP/QP solution algorithms".

initProbCPLEX *Create a CPLEX Problem Object in the CPLEX Environment*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXcreateprob`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
initProbCPLEX(env, pname = "CPLEX_PROB", ptrtype = "cplex_prob")
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
pname	A single character string containing the name of the problem object.
ptrtype	A name for the pointer object.

Details

Interface to the C function CPXcreate which calls the CPLEX function CPXcreateprob.

Value

If successful, a pointer to the CPLEX problem object is returned (an instance of class "cplexPtr"), otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[delProbCPLEX](#)

lpoptCPLEX

Find a Solution to a Problem Using One of the CPLEX Linear Optimizers

Description

Low level interface function to the IBM ILOG CPLEX function CPXlpopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
lpoptCPLEX(env, lp)
```

Arguments

env An object of class "cplexPtr" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by [initProbCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function lpopt which calls the CPLEX function CPXlpopt.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[solnInfoCPLEX](#), [getStatCPLEX](#), [solutionCPLEX](#)

mipoptCPLEX

Find a Solution to a Mixed Integer Program

Description

Low level interface function to the IBM ILOG CPLEX function CPXmipopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
mipoptCPLEX(env, lp)
```

Arguments

env An object of class "cplexPtr" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by [initProbCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function mipopt which calls the CPLEX function CPXmipopt.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[solnInfoCPLEX](#), [getStatCPLEX](#), [solutionCPLEX](#)

newColsCPLEX

Add Empty Columns to a Specified CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXnewcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
newColsCPLEX(env, lp, ncols,
             obj = NULL, lb = NULL, ub = NULL,
             xtype = NULL, cnames = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
ncols	Number of variables to add.
obj	Objective function coefficients.
lb	Lower bounds on the new variables.
ub	Upper bounds on the new variables.
xctype	Type of the new variables.
cnames	Names of the new variables.

Details

Interface to the C function newCols which calls the CPLEX function CPXnewcols.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#) section “Variable types for ctype array”.

 newRowsCPLEX

Add Empty Constraints to a Specified CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXnewrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
newRowsCPLEX(env, lp,
             nrows, rhs = NULL, sense = NULL,
             rngval = NULL, rnames = NULL)
```

Arguments

env	An object of class " cplexPtr " as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class " cplexPtr " as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
nrows	Number of new rows.
rhs	Right hand side term for each new constraint.
sense	Sense of each new constraint (see IBM ILOG CPLEX documentation for possible values).
rngval	Range values for each new constraint.
rnames	Names for the new rows.

Details

Interface to the C function newRows which calls the CPLEX function CPXnewrows.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

objSaCplex	<i>Access Upper and Lower Sensitivity Ranges for Objective Function Coefficients</i>
------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXobjsa. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
objSaCplex(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCplex</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCplex</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	Beginning of the range of ranges to be returned.
end	End of the range of ranges to be returned.

Details

Interface to the C function objSa which calls the CPLEX function CPXobjsa.

Value

If successful a list is returned:

lower	lower range values
upper	upper range values

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

openEnvCPLEX	<i>Initialize a CPLEX Environment</i>
--------------	---------------------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXopenCPLEX. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
openEnvCPLEX(ptrtype = "cplex_env")
```

Arguments

ptrtype A name for the pointer object.

Details

Interface to the C function openEnv which calls the CPLEX function CPXopenCPLEX.

Value

If successful, a pointer to the CPLEX environment is returned (an instance of class "`cplexPtr`"), otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

`closeEnvCPLEX`

`openFileCPLEX`*Open a File*

Description

Low level interface function to the IBM ILOG CPLEX function CPXfopen. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXfopen has been removed.

Usage

```
openFileCPLEX(fname, ftype = "w", ptrtype = "cplex_file")
```

Arguments

<code>fname</code>	Character string giving the file name to be opened.
<code>ftype</code>	Character string according to the syntax of the standard C function fopen.
<code>ptrtype</code>	A name for the pointer object.

Details

Interface to the C function `cplexfopen` which calls the CPLEX function `CPXfopen`.

Value

A pointer to the log file (an instance of class "`cplexPtr`") or NULL.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[closeFileCPLEX](#), [fileputCPLEX](#)

openProbCPLEX	<i>Create new CPLEX Environment And New CPLEX Problem Object</i>
---------------	--

Description

The function openProbCPLEX creates a new CPLEX environment and a new CPLEX problem object.

Usage

```
openProbCPLEX(pname = "CPLEX_PROB",  
              ptrtypeENV = "cplex_env",  
              ptrtypePROB = "cplex_prob")
```

Arguments

pname	A single character string containing the name of the problem object.
ptrtypeENV	A name for the IBM ILOG CPLEX environment pointer object.
ptrtypePROB	A name for the IBM ILOG CPLEX problem pointer object.

Details

Interface to the C functions openEnv and initProb calling CPLEX functions CPXopenCPLEX and CPXcreateprob.

Value

env	A pointer to the CPLEX environment as returned by openEnvCPLEX .
lp	A pointer to the CPLEX problem object as returned by initProbCPLEX .

If openEnvCPLEX() fails, env will be of class "cplexError" and lp will be NULL. Each list element is an object of class "cplexPtr".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[closeProbCPLEX](#), [openEnvCPLEX](#), [initProbCPLEX](#)

`ordWriteCPLEX`*Write Priority Order to ORD File*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXordwrite`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
ordWriteCPLEX(env, lp, fname)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>fname</code>	Filename.

Details

Interface to the C function `ordWrite` which calls the CPLEX function `CPXordwrite`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

preslvWriteCPLEX *Write a Presolved Version of the Problem to File*

Description

Low level interface function to the IBM ILOG CPLEX function CPXpreslvwrite. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
preslvWriteCPLEX(env, lp, fname)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
fname	Single character value giving the file name to write to.

Details

Interface to the C function preslvWrite which calls the CPLEX function CPXpreslvwrite.

Value

If successful a dingle numeric value containing the objective value difference between the original problem and the presolved problem, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[readCopyProbCPLEX](#)

presolveCPLEX	<i>Perform Presolve</i>
---------------	-------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXpresolve. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
presolveCPLEX(env, lp, method)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
method	A single integer value specifying the optimization algorithm to be used to solve the problem after the presolve is completed.

Details

Interface to the C function `presolve` which calls the CPLEX function `CPXpresolve`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

`cplexConstants` section "LP/QP solution algorithms".

primoptCPLEX

Find a Solution to a Problem Using the Primal Simplex Method

Description

Low level interface function to the IBM ILOG CPLEX function CPXprimopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
primoptCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function primopt which calls the CPLEX function CPXprimopt.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[solnInfoCPLEX](#), [getStatCPLEX](#), [solutionCPLEX](#)

printTerminateCPLEX *Print Termination Signal*

Description

The function `chgTerminateCPLEX` prints termination signal.

Usage

```
printTerminateCPLEX(env)
```

Arguments

`env` An object of class "`cplexPtr`" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

Value

NULL

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[setTerminateCPLEX](#), [delTerminateCPLEX](#), [chgTerminateCPLEX](#)

qpoptCPLEX *Find a Solution to a Continuous Quadratic Program*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXqpopt`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
qpoptCPLEX(env, lp)
```


Arguments

- env An object of class "cplexPtr" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.
- lp An object of class "cplexPtr" as returned by [initProbCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `qpopt` which calls the CPLEX function `CPXqpopt`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[solnInfoCPLEX](#), [getStatCPLEX](#), [solutionCPLEX](#)

readCopyBaseCPLEX	<i>Read Basis From a BAS File and Copy it Into a CPLEX Problem Object</i>
-------------------	---

Description

Low level interface function to the IBM ILOG CPLEX function `CPXreadcopybase`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
readCopyBaseCPLEX(env, lp, fname)
```

Arguments

- env An object of class "cplexPtr" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.
- lp An object of class "cplexPtr" as returned by [initProbCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX problem object.
- fname Single character value giving the filename to read from.

Details

Interface to the C function readCopyBase which calls the CPLEX function CPXreadcopybase.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

readCopyMIPstartsCPLEX

Read a File in the Format MST

Description

Low level interface function to the IBM ILOG CPLEX function CPXreadcopymipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
readCopyMIPstartsCPLEX(env, lp, fname)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
fname	Name of the file to read from.s

Details

Interface to the C function readCopyMIPstarts which calls the CPLEX function CPXreadcopymipstarts.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

readCopyOrderCPLEX *Read ORD File*

Description

Low level interface function to the IBM ILOG CPLEX function CPXreadcopyorder. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
readCopyOrderCPLEX(env, lp, fname)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
fname	Single character value giving the filename to read from.

Details

Interface to the C function `readCopyOrder` which calls the CPLEX function `CPXreadcopyorder`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

readCopyParmCPLEX *Reads Parameter Names And Settings From a File*

Description

Low level interface function to the IBM ILOG CPLEX function CPXreadcopyparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
readCopyParmCPLEX(env, fname)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
fname	Filename.

Details

Interface to the C function readCopyParm which calls the CPLEX function CPXreadcopyparam.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[cplexConstants](#)

readCopyProbCPLEX *Read an MPS, LP, or SAV File Into an Existing CPLEX Problem Object*

Description

Low level interface function to the IBM ILOG CPLEX function CPXreadcopyprob. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
readCopyProbCPLEX(env, lp, fname, ftype = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
fname	Single character value giving the filename to read from.
ftype	Single character value giving the type of the file to read from.

Details

Interface to the C function `readCopyProb` which calls the CPLEX function `CPXreadcopyprob`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

readCopySolCplex *Reads a Solution From a SOL Format File*

Description

Low level interface function to the IBM ILOG CPLEX function CPXreadcopySol. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
readCopySolCplex(env, lp, fname)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCplex</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCplex</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
fname	Single character value giving the filename to read from.

Details

Interface to the C function `readCopySol` which calls the CPLEX function `CPXreadcopySol`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

refineConflictCPLEX *Identify a Minimal Conflict for the Infeasibility of the Linear Constraints and the Variable Bounds*

Description

Low level interface function to the IBM ILOG CPLEX function CPXrefineconflict. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
refineConflictCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `refineConflict` which calls the CPLEX function `CPXrefineconflict`.

Value

If successful a list is returned:

confnumrows number of linear constraints in the conflict

confnumcols number of variable bounds in the conflict

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[getConflictCPLEX](#)

refineConflictExtCPLEX

Identify a Minimal Conflict

Description

Low level interface function to the IBM ILOG CPLEX function CPXrefineconflicttext. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
refineConflictExtCPLEX(env, lp, grpcnt, concnt,  
                       grppref, grpbeg, grpind, grptype)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
grpcnt	The number of constraint groups to be considered.
concnt	Length of arrays grpind and grptype.
grppref	Preferences for the groups.
grpbeg	The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.
grpind	The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.
grptype	The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.

Details

Interface to the C function `refineConflictExt` which calls the CPLEX function `CPXrefineconflicttext`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

`refineMIPstartConflictCPLEX`*Refine a Conflict in Order to Determine Why a Given MIP Start is Not Feasible*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXrefinemipstartconflict`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
refineMIPstartConflictCPLEX(env, lp, mipstartindex)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>mipstartindex</code>	The index of the MIP start.

Details

Interface to the C function `refineMIPstartConflict` which calls the CPLEX function `CPXrefinemipstartconflict`.

Value

If successful a list is returned:

<code>confnumrows</code>	number of linear constraints in the conflict
<code>confnumcols</code>	number of variable bounds in the conflict

otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[getConflictCPLEX](#)

 refineMIPstartConflictExtCPLEX

Identify a Minimal Conflict

Description

Low level interface function to the IBM ILOG CPLEX function CPXrefinemipstartconflicttext. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
refineMIPstartConflictExtCPLEX(env, lp, mipstartindex, grpcnt, concnt,
                               grppref, grpbeg, grpind, grptype)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
mipstartindex	The index of the MIP start.
grpcnt	The number of constraint groups to be considered.
concnt	Length of arrays grpind and grptype.
grppref	Preferences for the groups.
grpbeg	The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.
grpind	The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.
grptype	The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.

Details

Interface to the C function refineMIPstartConflictExt which calls the CPLEX function CPXrefinemipstartconflicttext.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

return_codeCPLEX	<i>Translates a IBM ILOG CPLEX Return Code into a Human Readable String</i>
------------------	---

Description

Translates a IBM ILOG CPLEX return code into a human readable string.

Usage

```
return_codeCPLEX(code)
```

Arguments

code	Return (error) code from IBM ILOG CPLEX.
------	--

Value

An error message string corresponding to an return (error) code.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

rhsSaCplex	<i>Access Upper and Lower Sensitivity Ranges for Righthand Side Values of a Range of Constraints</i>
------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXrhssa. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
rhsSaCplex(env, lp, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCplex</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCplex</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
begin	Beginning of the range of ranges to be returned.
end	End of the range of ranges to be returned.

Details

Interface to the C function rhsSa which calls the CPLEX function CPXrhssa.

Value

If successful a list is returned:

lower	righthand side lower range values
upper	righthand side upper range values

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

setDbiParamCPLEX	<i>Set the Value of a CPLEX Parameter of Type Double</i>
------------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetdblparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
setDbiParamCPLEX(env, parm, value)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.
value	The new value of the parameter.

Details

Interface to the C function setDbiParam which calls the CPLEX function CPXsetdblparam.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

setDefaultParmCPLEX *Reset All CPLEX Parameters And Settings to Default Values*

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetdefaults. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
setDefaultParmCPLEX(env)
```

Arguments

env An object of class "cplexPtr" as returned by [openEnvCPLEX](#). This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function setDefaultParm which calls the CPLEX function CPXsetdefaults.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

setIntParmCPLEX	<i>Set the Value of a CPLEX Parameter of Type CPXINT</i>
-----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetintparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
setIntParmCPLEX(env, parm, value)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.
value	The new value of the parameter (integer value).

Details

Interface to the C function setIntParm which calls the CPLEX function CPXsetintparam.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

setLogFileCPLEX	<i>Modifies the log file to which Messages are Written</i>
-----------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetlogfile. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXsetlogfile has been removed.

Usage

```
setLogFileCPLEX(env, cpfile = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
cpfile	A pointer to a file as returned by openFileCPLEX .

Details

Interface to the C function getLogFile which calls the CPLEX function CPXgetlogfile.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[getLogFileCPLEX](#)

setLogFileNameCPLEX *Set and open a log file*

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetlogfile. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality for IBM ILOG CPLEX < 12.8.0, where CPXsetlogfile was not included.

Usage

```
setLogFileNameCPLEX(env, filename = "cpx.log", mode = "w")
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
filename	The name of the log file to open.
mode	The mode in which CPLEX should open the file. The specification is the same as that for the C library function fopen. For example, use a quoted character, such as "w" to write or "a" to append. Make sure you open the file for writing; otherwise, CPLEX writes nothing to the log file, and CPLEX can produce an error every time it attempts to write. If filename is NULL, then this argument is ignored and can be NULL, too.

Details

Interface to the C function setLogFileName which calls the CPLEX function CPXsetlogfile.

Value

Zero if successful, otherwise nonzero.

Author(s)

Mayo Roettger <mayo.roettger@hhu.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

setLongParmCPLEX *Set the Value of a Parameter of Type CPXLONG*

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetlongparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
setLongParmCPLEX(env, parm, value)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.
value	New value for the parameter.

Details

Interface to the C function `setLongParm` which calls the CPLEX function `CPXsetlongparam`.

Value

Zero if successful, otherwise nonzero.

Note

In order to transfer a 64 bit integer value to `CPXsetlongparam`, datatype `numeric` is used. Parameter value is a numeric value.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[setIntParmCPLEX](#), [cplexConstants](#)

setObjDirCPLEX	<i>Change the Sense of the Optimization for a Problem</i>
----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgobjsen. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
setObjDirCPLEX(env, lp, lpdire)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
lpdire	A single integer value specifying the sense of the problem.

Details

Interface to the C function setObjDir which calls the CPLEX function CPXchgobjsen.

Value

NULL

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#) section "Generic constants".

setStrParmCPLEX *Set the Value of a CPLEX String Parameter*

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetstrparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
setStrParmCPLEX(env, parm, value)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
parm	Constant or reference number of the desired parameter.
value	The new value of the parameter (character value).

Details

Interface to the C function setStrParm which calls the CPLEX function CPXsetstrparam.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#)

setTerminateCPLEX *Release Termination Signal*

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetterminate with argument terminate_p set to NULL. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
setTerminateCPLEX(env, ptrtype = "cplex_term")
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
ptrtype	A name for the pointer object.

Details

Interface to the C function delTerminate which calls the CPLEX function CPXsetterminate with argument terminate_p set to NULL.

Value

If successful, a pointer to a termination signal is returned, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[delTerminateCPLEX](#), [printTerminateCPLEX](#), [chgTerminateCPLEX](#)

siftoptCPLEX	<i>Solve a Reduced Model</i>
--------------	------------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXsiftopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
siftoptCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `siftopt` which calls the CPLEX function `CPXsiftopt`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

solnInfoCPLEX	<i>Access Solution Information</i>
---------------	------------------------------------

Description

Low level interface function to the IBM ILOG CPLEX function CPXsolninfo. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
solnInfoCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `solnInfo` which calls the CPLEX function `CPXsolninfo`.

Value

If successful a list is returned:

method	Integer value specifying the method to produce the current solution.
type	Integer value specifying the type of current solution.
primal_feasible	Integer value specifying if the current solution is known to be primal feasible.
dual_feasible	Integer value specifying if the current solution is known to be dual feasible.

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

`cplexConstants`, `solutionCPLEX`

 solutionCPLEX

Access Solution Values Produced by Optimization Routines

Description

Low level interface function to the IBM ILOG CPLEX function CPXsolution. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
solutionCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `solution` which calls the CPLEX function CPXsolution.

Value

If successful a list is returned:

lpstat	result of the optimization
objval	objective function value
x	values of the variables for the problem
pi	values of the dual variables
slack	values of the slack or surplus variables
dj	values of the reduced costs

otherwise an instance of class "[cplexError](#)".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[cplexConstants](#), [solnInfoCPLEX](#)

`solWriteCPLEX`*Write a Solution File*

Description

Low level interface function to the IBM ILOG CPLEX function `CPXsolwrite`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
solWriteCPLEX(env, lp, fname)
```

Arguments

<code>env</code>	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
<code>lp</code>	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
<code>fname</code>	Single character value giving the filename to write to.

Details

Interface to the C function `solWrite` which calls the CPLEX function `CPXsolwrite`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

`getProbTypeCPLEX`, `cplexConstants` section "Problem Types".

status_codeCPLEX	<i>Translates an IBM ILOG CPLEX Status Value into a Human Readable String</i>
------------------	---

Description

Translates a IBM ILOG CPLEX status code into a human readable string.

Usage

```
status_codeCPLEX(env, code)
```

Arguments

env	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
code	Status code from IBM ILOG CPLEX as returned by <code>getStatCPLEX</code> .

Value

A character string orresponding to the value of an IBM ILOG CPLEX status code as returned by `getStatCPLEX`.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[cplexConstants](#), [getStatStrCPLEX](#)

tightenBndsCPLEX *Change the Lower or Upper Bounds on a Set of Variables of a Problem*

Description

Low level interface function to the IBM ILOG CPLEX function CPXtightenbds. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
tightenBndsCPLEX(env, lp, ncols, ind, lu, bd)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
ncols	Number of bounds to be changed.
ind	Indices of bounds to be changed.
lu	A character vector, specifying whether an entry in bd is a upper or a lower bound on variable ind[j].
bd	Values of the lower or upper bounds of the variables present in ind.

Details

Interface to the C function tightenBnds which calls the CPLEX function CPXtightenbds.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

See Also

[chgBndsCPLEX](#)

tuneParmCPLEX	<i>Tune Parameters of the Environment For Improved Optimizer Performance</i>
---------------	--

Description

Low level interface function to the IBM ILOG CPLEX function CPXtuneparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
tuneParmCPLEX(env, lp,
              nIntP = 0, intP = NULL, intPv = NULL,
              nDb1P = 0, db1P = NULL, db1Pv = NULL,
              nStrP = 0, strP = NULL, strPv = NULL)
```

Arguments

env	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class " <code>cplexPtr</code> " as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
nIntP	Number of integer parameters to be fixed during tuning.
intP	Parameter numbers of the integer parameters which remain fixed.
intPv	Values for the parameters listed in intP.
nDb1P	Number of double parameters to be fixed during tuning.
db1P	Parameter numbers of the double parameters which remain fixed.
db1Pv	Values for the parameters listed in db1P.
nStrP	Number of string parameters to be fixed during tuning.
strP	Parameter numbers of the string parameters which remain fixed.
strPv	Values for the parameters listed in strP.

Details

Interface to the C function tuneParam which calls the CPLEX function CPXtuneparam.

Value

Zero if successful, otherwise an instance of class "`cplexError`".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

unscaleProbCPLEX *Remove Any Scaling Applied to the Resident Problem*

Description

Low level interface function to the IBM ILOG CPLEX function CPXunscalereprob. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
unscaleProbCPLEX(env, lp)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `unscalereprob` which calls the CPLEX function `CPXunscalereprob`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

writeMIPstartsCPLEX *Write a Range of MIP Starts to a File in MST Format*

Description

Low level interface function to the IBM ILOG CPLEX function CPXwritemipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
writeMIPstartsCPLEX(env, lp, fname, begin, end)
```

Arguments

env	An object of class "cplexPtr" as returned by openEnvCPLEX . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by initProbCPLEX . This is basically a pointer to an IBM ILOG CPLEX problem object.
fname	Filename to write to.
begin	An integer specifying the beginning of the range of MIP starts to be written.
end	An integer specifying the end of the range of MIP starts to be written.

Details

Interface to the C function `writemipstarts` which calls the CPLEX function `CPXwritemipstarts`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

writeParmCPLEX	<i>Write Names and Current Settings of CPLEX Parameters to File</i>
----------------	---

Description

Low level interface function to the IBM ILOG CPLEX function `CPXwriteparam`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
writeParmCPLEX(env, fname)
```

Arguments

env	An object of class " <code>cplexPtr</code> " as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
fname	Filename.

Details

Interface to the C function `writeParm` which calls the CPLEX function `CPXwriteparam`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSA5P>.

See Also

[cplexConstants](#)

writeProbCPLEX *Write a CPLEX Problem Object to File*

Description

Low level interface function to the IBM ILOG CPLEX function CPXwriteprob. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
writeProbCPLEX(env, lp, fname, ftype = NULL)
```

Arguments

env	An object of class "cplexPtr" as returned by <code>openEnvCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX environment.
lp	An object of class "cplexPtr" as returned by <code>initProbCPLEX</code> . This is basically a pointer to an IBM ILOG CPLEX problem object.
fname	Single character value giving the file name to write to.
ftype	Single character value giving the type of the file to write to.

Details

Interface to the C function `writeProb` which calls the CPLEX function `CPXwriteprob`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at <https://www.ibm.com/support/knowledgecenter/SSSA5P>.

Index

* optimize

- addColsCPLEX, 7
- addFpDestCPLEX, 9
- addIndConstrCPLEX, 10
- addMIPstartsCPLEX, 11
- addQConstrCPLEX, 12
- addRowsCPLEX, 13
- baroptCPLEX, 14
- baseWriteCPLEX, 15
- basicPresolveCPLEX, 16
- boundSaCPLEX, 17
- checkAddColsCPLEX, 18
- checkAddRowsCPLEX, 19
- checkChgCoefListCPLEX, 20
- checkCopyColTypeCPLEX, 21
- checkCopyLpCPLEX, 22
- checkCopyLpwNamesCPLEX, 23
- checkCopyQPsepCPLEX, 25
- checkCopyQuadCPLEX, 26
- checkValsCPLEX, 27
- chgBndsCPLEX, 28
- chgCoefCPLEX, 29
- chgCoefListCPLEX, 30
- chgColNameCPLEX, 31
- chgColsBndsCPLEX, 32
- chgColTypeCPLEX, 33
- chgMIPstartsCPLEX, 34
- chgNameCPLEX, 35
- chgObjCPLEX, 36
- chgProbNameCPLEX, 37
- chgProbTypeCPLEX, 38
- chgQPcoefCPLEX, 39
- chgRhsCPLEX, 40
- chgRngValCPLEX, 41
- chgRowNameCPLEX, 42
- chgSenseCPLEX, 43
- chgTerminateCPLEX, 44
- cleanupCoefCPLEX, 44
- cloneProbCPLEX, 45
- closeEnvCPLEX, 46
- closeFileCPLEX, 47
- closeProbCPLEX, 48
- cLpWriteCPLEX, 49
- completeIpCPLEX, 50
- copyBaseCPLEX, 51
- copyColTypeCPLEX, 52
- copyLpCPLEX, 53
- copyLpwNamesCPLEX, 54
- copyObjNameCPLEX, 55
- copyOrderCPLEX, 56
- copyQPsepCPLEX, 57
- copyQuadCPLEX, 58
- copyStartCPLEX, 59
- cplexAPI-package, 6
- cplexConstants, 60
- cplexError-class, 82
- cplexPtr-class, 83
- delColsCPLEX, 84
- delFpDestCPLEX, 85
- delIndConstrsCPLEX, 86
- delMIPstartsCPLEX, 87
- delNamesCPLEX, 88
- delProbCPLEX, 89
- delQConstrsCPLEX, 90
- delRowsCPLEX, 91
- delSetColsCPLEX, 92
- delSetRowsCPLEX, 93
- delTerminateCPLEX, 94
- disconnectChannelCPLEX, 95
- dualoptCPLEX, 96
- dualWriteCPLEX, 97
- feasOptCPLEX, 98
- fileputCPLEX, 99
- flushChannelCPLEX, 100
- flushStdChannelsCPLEX, 101
- freePresolveCPLEX, 102
- getBaseCPLEX, 103
- getBestObjValCPLEX, 104

getChannelsCPLEX, 105
getChgParmCPLEX, 106
getCoefCPLEX, 107
getColIndexCPLEX, 108
getColInfeasCPLEX, 109
getColNameCPLEX, 110
getColsCPLEX, 111
getColTypeCPLEX, 112
getConflictCPLEX, 113
getConflictExtCPLEX, 114
getCutoffCPLEX, 115
getDb1ParmCPLEX, 116
getDb1QualCPLEX, 117
getDb1CntCPLEX, 118
getDjCPLEX, 119
getErrorStrCPLEX, 120
getGradCPLEX, 121
getIndConstrCPLEX, 122
getInfoDb1ParmCPLEX, 123
getInfoIntParmCPLEX, 124
getInfoLongParmCPLEX, 125
getInfoStrParmCPLEX, 126
getIntParmCPLEX, 127
getIntQualCPLEX, 128
getItCntCPLEX, 129
getLogFileCPLEX, 130
getLogFileNameCPLEX, 131
getLongParmCPLEX, 132
getLowBndsIdsCPLEX, 133
getLowerBndsCPLEX, 134
getMethodCPLEX, 135
getMIPrelGapCPLEX, 136
getMIPstartIndexCPLEX, 137
getMIPstartNameCPLEX, 138
getMIPstartsCPLEX, 139
getNumColsCPLEX, 140
getNumMIPstartsCPLEX, 141
getNumNnzCPLEX, 142
getNumQConstrsCPLEX, 143
getNumQPnzCPLEX, 144
getNumQuadCPLEX, 145
getNumRowsCPLEX, 146
getObjCPLEX, 147
getObjDirCPLEX, 148
getObjNameCPLEX, 149
getObjOffsetCPLEX, 150
getObjValCPLEX, 151
getOrderCPLEX, 152
getParmHierNameCPLEX, 153
getParmNameCPLEX, 154
getParmNumCPLEX, 155
getParmTypeCPLEX, 156
getParmValCPLEX, 157
getPhase1CntCPLEX, 157
getPiCPLEX, 158
getPreStatCPLEX, 159
getProbNameCPLEX, 160
getProbTypeCPLEX, 161
getProbVarCPLEX, 162
getQConstrCPLEX, 163
getQPcoefCPLEX, 164
getQuadCPLEX, 165
getRedLpCPLEX, 166
getRhsCPLEX, 167
getRngValCPLEX, 168
getRowIndexCPLEX, 169
getRowInfeasCPLEX, 170
getRowNameCPLEX, 171
getRowsCPLEX, 172
getSenseCPLEX, 173
getSiftItCntCPLEX, 174
getSiftPase1CntCPLEX, 175
getSlackCPLEX, 176
getStatCPLEX, 177
getStatStrCPLEX, 178
getStrParmCPLEX, 179
getSubMethodCPLEX, 180
getSubStatCPLEX, 181
getTimeCPLEX, 182
getUppBndsIdsCPLEX, 183
getUpperBndsCPLEX, 184
getVersionCPLEX, 185
hybbaroptCPLEX, 185
hybnetoptCPLEX, 186
initProbCPLEX, 187
lpoptCPLEX, 188
mipoptCPLEX, 189
newColsCPLEX, 190
newRowsCPLEX, 191
objSaCPLEX, 192
openEnvCPLEX, 193
openFileCPLEX, 194
openProbCPLEX, 195
ordWriteCPLEX, 196
preslvWriteCPLEX, 197
presolveCPLEX, 198

- primoptCPLEX, 199
- printTerminateCPLEX, 200
- qpoptCPLEX, 200
- readCopyBaseCPLEX, 201
- readCopyMIPstartsCPLEX, 202
- readCopyOrderCPLEX, 203
- readCopyParmCPLEX, 204
- readCopyProbCPLEX, 205
- readCopySolCPLEX, 206
- refineConflictCPLEX, 207
- refineConflictExtCPLEX, 208
- refineMIPstartConflictCPLEX, 209
- refineMIPstartConflictExtCPLEX, 210
- return_codeCPLEX, 211
- rhsSaCPLEX, 212
- setDb1ParmCPLEX, 213
- setDefaultParmCPLEX, 214
- setIntParmCPLEX, 215
- setLogFileCPLEX, 216
- setLogFileNameCPLEX, 217
- setLongParmCPLEX, 218
- setObjDirCPLEX, 219
- setStrParmCPLEX, 220
- setTerminateCPLEX, 221
- siftoptCPLEX, 222
- solnInfoCPLEX, 223
- solutionCPLEX, 224
- solWriteCPLEX, 225
- status_codeCPLEX, 226
- tightenBndsCPLEX, 227
- tuneParmCPLEX, 228
- unscaleProbCPLEX, 229
- writeMIPstartsCPLEX, 230
- writeParmCPLEX, 231
- writeProbCPLEX, 232
- * **package**
 - cp1exAPI-package, 6
- addColsCPLEX, 7, 14, 19
- addFpDestCPLEX, 9, 85
- addIndConstrCPLEX, 10
- addMIPstartsCPLEX, 11
- addQConstrCPLEX, 12
- addRowsCPLEX, 9, 13, 20
- baroptCPLEX, 14
- baseWriteCPLEX, 15
- basicPresolveCPLEX, 16
- boundSaCPLEX, 17
- checkAddColsCPLEX, 9, 18
- checkAddRowsCPLEX, 14, 19
- checkChgCoefListCPLEX, 20
- checkCopyColTypeCPLEX, 21
- checkCopyLpCPLEX, 22
- checkCopyLpwNamesCPLEX, 23
- checkCopyQPsepCPLEX, 25
- checkCopyQuadCPLEX, 26
- checkValsCPLEX, 27
- chgBndsCPLEX, 28, 32, 227
- chgCoefCPLEX, 29, 39
- chgCoefListCPLEX, 21, 30
- chgColNameCPLEX, 31
- chgColsBndsCPLEX, 32
- chgColTypeCPLEX, 33
- chgMIPstartsCPLEX, 34
- chgNameCPLEX, 35
- chgObjCPLEX, 29, 36
- chgProbNameCPLEX, 37
- chgProbTypeCPLEX, 38, 161
- chgQPcoefCPLEX, 39
- chgRhsCPLEX, 29, 40
- chgRngValCPLEX, 14, 29, 41
- chgRowNameCPLEX, 42
- chgSenseCPLEX, 43
- chgTerminateCPLEX, 44, 94, 200, 221
- cleanupCoefCPLEX, 44
- cloneProbCPLEX, 45
- closeEnvCPLEX, 46, 193
- closeFileCPLEX, 47, 99, 194
- closeProbCPLEX, 48, 195
- clpWriteCPLEX, 49
- complete1pCPLEX, 50
- constantsCPLEX (cplexConstants), 60
- copyBaseCPLEX, 51
- copyColTypeCPLEX, 22, 52
- copyLpCPLEX, 14, 23, 53
- copyLpwNamesCPLEX, 24, 54
- copyObjNameCPLEX, 55
- copyOrderCPLEX, 56
- copyQPsepCPLEX, 25, 57
- copyQuadCPLEX, 26, 58
- copyStartCPLEX, 59
- cplex_Constants (cplexConstants), 60
- cp1exAPI (cp1exAPI-package), 6
- cp1exAPI-package, 6

- cplexConstants, [38](#), [60](#), [106](#), [112](#), [114](#), [116](#), [117](#), [124–127](#), [131](#), [132](#), [135](#), [148](#), [152–157](#), [160](#), [161](#), [177](#), [179](#), [180](#), [186](#), [187](#), [191](#), [196](#), [198](#), [204](#), [211](#), [213–215](#), [217–220](#), [223–226](#), [231](#)
- cplexError, [16](#), [17](#), [46](#), [47](#), [97](#), [103–117](#), [119](#), [121–125](#), [127](#), [128](#), [130](#), [132–134](#), [136–139](#), [147](#), [149–152](#), [156](#), [159](#), [160](#), [162–173](#), [176](#), [182–184](#), [188](#), [192](#), [193](#), [195](#), [197](#), [207](#), [209](#), [212](#), [221](#), [223](#), [224](#), [228](#)
- cplexError (cplexError-class), [82](#)
- cplexError-class, [82](#)
- cplexPointer (cplexPtr-class), [83](#)
- cplexPointer, cplexPtr-method (cplexPtr-class), [83](#)
- cplexPtr, [8–23](#), [25–46](#), [48–59](#), [84–98](#), [100–210](#), [212–232](#)
- cplexPtr (cplexPtr-class), [83](#)
- cplexPtr-class, [83](#)
- cplexPtrType (cplexPtr-class), [83](#)
- cplexPtrType, cplexPtr-method (cplexPtr-class), [83](#)
- cplexPtrType<- (cplexPtr-class), [83](#)
- cplexPtrType<- , cplexPtr-method (cplexPtr-class), [83](#)
- CPX_ALG_AUTOMATIC (cplexConstants), [60](#)
- CPX_ALG_BAROPT (cplexConstants), [60](#)
- CPX_ALG_BARRIER (cplexConstants), [60](#)
- CPX_ALG_CONCURRENT (cplexConstants), [60](#)
- CPX_ALG_DUAL (cplexConstants), [60](#)
- CPX_ALG_FEASOPT (cplexConstants), [60](#)
- CPX_ALG_MIP (cplexConstants), [60](#)
- CPX_ALG_NET (cplexConstants), [60](#)
- CPX_ALG_NONE (cplexConstants), [60](#)
- CPX_ALG_PIVOT (cplexConstants), [60](#)
- CPX_ALG_PIVOTIN (cplexConstants), [60](#)
- CPX_ALG_PIVOTOUT (cplexConstants), [60](#)
- CPX_ALG_PRIMAL (cplexConstants), [60](#)
- CPX_ALG_ROBUST (cplexConstants), [60](#)
- CPX_ALG_SIFTING (cplexConstants), [60](#)
- CPX_AT_LOWER (cplexConstants), [60](#)
- CPX_AT_UPPER (cplexConstants), [60](#)
- CPX_BARORDER_AMD (cplexConstants), [60](#)
- CPX_BARORDER_AMF (cplexConstants), [60](#)
- CPX_BARORDER_AUTO (cplexConstants), [60](#)
- CPX_BARORDER_ND (cplexConstants), [60](#)
- CPX_BASIC (cplexConstants), [60](#)
- CPX_BASIC_SOLN (cplexConstants), [60](#)
- CPX_BINARY (cplexConstants), [60](#)
- CPX_BRANCH_DOWN (cplexConstants), [60](#)
- CPX_BRANCH_GLOBAL (cplexConstants), [60](#)
- CPX_BRANCH_UP (cplexConstants), [60](#)
- CPX_BRDIR_AUTO (cplexConstants), [60](#)
- CPX_BRDIR_DOWN (cplexConstants), [60](#)
- CPX_BRDIR_UP (cplexConstants), [60](#)
- CPX_CALLBACKCONTEXT_CANDIDATE (cplexConstants), [60](#)
- CPX_CALLBACKCONTEXT_GLOBAL_PROGRESS (cplexConstants), [60](#)
- CPX_CALLBACKCONTEXT_LOCAL_PROGRESS (cplexConstants), [60](#)
- CPX_CALLBACKCONTEXT_RELAXATION (cplexConstants), [60](#)
- CPX_CALLBACKCONTEXT_THREAD_DOWN (cplexConstants), [60](#)
- CPX_CALLBACKCONTEXT_THREAD_UP (cplexConstants), [60](#)
- CPX_CON_ABS (cplexConstants), [60](#)
- CPX_CON_DISJCST (cplexConstants), [60](#)
- CPX_CON_INDDISJCST (cplexConstants), [60](#)
- CPX_CON_INDICATOR (cplexConstants), [60](#)
- CPX_CON_LAST_CONTYPE (cplexConstants), [60](#)
- CPX_CON_LINEAR (cplexConstants), [60](#)
- CPX_CON_LOWER_BOUND (cplexConstants), [60](#)
- CPX_CON_MAXEXPR (cplexConstants), [60](#)
- CPX_CON_MINEXPR (cplexConstants), [60](#)
- CPX_CON_PWL (cplexConstants), [60](#)
- CPX_CON_QUADRATIC (cplexConstants), [60](#)
- CPX_CON_SETVAR (cplexConstants), [60](#)
- CPX_CON_SETVARCARD (cplexConstants), [60](#)
- CPX_CON_SETVARDOMAIN (cplexConstants), [60](#)
- CPX_CON_SETVAREQ (cplexConstants), [60](#)
- CPX_CON_SETVARINTERSECT (cplexConstants), [60](#)
- CPX_CON_SETVARINTERSECTION (cplexConstants), [60](#)
- CPX_CON_SETVARMAX (cplexConstants), [60](#)
- CPX_CON_SETVARMEMBER (cplexConstants), [60](#)
- CPX_CON_SETVARMIN (cplexConstants), [60](#)
- CPX_CON_SETVARNEQ (cplexConstants), [60](#)
- CPX_CON_SETVARNEQCST (cplexConstants), [60](#)

- CPX_CON_SETVARNULLINTERSECT
(cplexConstants), 60
- CPX_CON_SETVARSUBSET (cplexConstants),
60
- CPX_CON_SETVARSUM (cplexConstants), 60
- CPX_CON_SETVARUNION (cplexConstants), 60
- CPX_CON_SOS (cplexConstants), 60
- CPX_CON_UPPER_BOUND (cplexConstants), 60
- CPX_CONFLICT_EXCLUDED (cplexConstants),
60
- CPX_CONFLICT_LB (cplexConstants), 60
- CPX_CONFLICT_MEMBER (cplexConstants), 60
- CPX_CONFLICT_POSSIBLE_LB
(cplexConstants), 60
- CPX_CONFLICT_POSSIBLE_MEMBER
(cplexConstants), 60
- CPX_CONFLICT_POSSIBLE_UB
(cplexConstants), 60
- CPX_CONFLICT_UB (cplexConstants), 60
- CPX_CONTINUOUS (cplexConstants), 60
- CPX_DPRIIND_AUTO (cplexConstants), 60
- CPX_DPRIIND_DEVEX (cplexConstants), 60
- CPX_DPRIIND_FULL (cplexConstants), 60
- CPX_DPRIIND_FULL_STEEP
(cplexConstants), 60
- CPX_DPRIIND_STEEP (cplexConstants), 60
- CPX_DPRIIND_STEEPQSTART
(cplexConstants), 60
- CPX_DUAL_OBJ (cplexConstants), 60
- CPX_EXACT_KAPPA (cplexConstants), 60
- CPX_FEASOPT_MIN_INF (cplexConstants), 60
- CPX_FEASOPT_MIN_QUAD (cplexConstants),
60
- CPX_FEASOPT_MIN_SUM (cplexConstants), 60
- CPX_FEASOPT_OPT_INF (cplexConstants), 60
- CPX_FEASOPT_OPT_QUAD (cplexConstants),
60
- CPX_FEASOPT_OPT_SUM (cplexConstants), 60
- CPX_FREE_SUPER (cplexConstants), 60
- CPX_IMPLIED_INTEGER_FEASIBLE
(cplexConstants), 60
- CPX_INFBOUND (cplexConstants), 60
- CPX_INTEGER (cplexConstants), 60
- CPX_INTEGER_FEASIBLE (cplexConstants),
60
- CPX_INTEGER_INFEASIBLE
(cplexConstants), 60
- CPX_KAPPA (cplexConstants), 60
- CPX_KAPPA_ATTENTION (cplexConstants), 60
- CPX_KAPPA_ILLPOSED (cplexConstants), 60
- CPX_KAPPA_MAX (cplexConstants), 60
- CPX_KAPPA_STABLE (cplexConstants), 60
- CPX_KAPPA_SUSPICIOUS (cplexConstants),
60
- CPX_KAPPA_UNSTABLE (cplexConstants), 60
- CPX_MAX (cplexConstants), 60
- CPX_MAX_COMP_SLACK (cplexConstants), 60
- CPX_MAX_DUAL_INFEAS (cplexConstants), 60
- CPX_MAX_DUAL_RESIDUAL (cplexConstants),
60
- CPX_MAX_INDSLACK_INFEAS
(cplexConstants), 60
- CPX_MAX_INT_INFEAS (cplexConstants), 60
- CPX_MAX_PI (cplexConstants), 60
- CPX_MAX_PRIMAL_INFEAS (cplexConstants),
60
- CPX_MAX_PRIMAL_RESIDUAL
(cplexConstants), 60
- CPX_MAX_PWSLACK_INFEAS
(cplexConstants), 60
- CPX_MAX_QCPRIMAL_RESIDUAL
(cplexConstants), 60
- CPX_MAX_QCSLACK (cplexConstants), 60
- CPX_MAX_QCSLACK_INFEAS
(cplexConstants), 60
- CPX_MAX_RED_COST (cplexConstants), 60
- CPX_MAX_SCALED_DUAL_INFEAS
(cplexConstants), 60
- CPX_MAX_SCALED_DUAL_RESIDUAL
(cplexConstants), 60
- CPX_MAX_SCALED_PI (cplexConstants), 60
- CPX_MAX_SCALED_PRIMAL_INFEAS
(cplexConstants), 60
- CPX_MAX_SCALED_PRIMAL_RESIDUAL
(cplexConstants), 60
- CPX_MAX_SCALED_RED_COST
(cplexConstants), 60
- CPX_MAX_SCALED_SLACK (cplexConstants),
60
- CPX_MAX_SCALED_X (cplexConstants), 60
- CPX_MAX_SLACK (cplexConstants), 60
- CPX_MAX_X (cplexConstants), 60
- CPX_MIN (cplexConstants), 60
- CPX_MIPEMPHASIS_BALANCED
(cplexConstants), 60
- CPX_MIPEMPHASIS_BESTBOUND

- (cplexConstants), 60
- CPX_MIPEMPHASIS_FEASIBILITY (cplexConstants), 60
- CPX_MIPEMPHASIS_HIDDENFEAS (cplexConstants), 60
- CPX_MIPEMPHASIS_OPTIMALITY (cplexConstants), 60
- CPX_MIPKAPPA_AUTO (cplexConstants), 60
- CPX_MIPKAPPA_FULL (cplexConstants), 60
- CPX_MIPKAPPA_OFF (cplexConstants), 60
- CPX_MIPKAPPA_SAMPLE (cplexConstants), 60
- CPX_MIPORDER_BOUNDS (cplexConstants), 60
- CPX_MIPORDER_COST (cplexConstants), 60
- CPX_MIPORDER_SCALED COST (cplexConstants), 60
- CPX_MIPSEARCH_AUTO (cplexConstants), 60
- CPX_MIPSEARCH_DYNAMIC (cplexConstants), 60
- CPX_MIPSEARCH_TRADITIONAL (cplexConstants), 60
- CPX_MIPSTART_AUTO (cplexConstants), 60
- CPX_MIPSTART_CHECKFEAS (cplexConstants), 60
- CPX_MIPSTART_REPAIR (cplexConstants), 60
- CPX_MIPSTART_SOLVEFIXED (cplexConstants), 60
- CPX_MIPSTART_SOLVEMIP (cplexConstants), 60
- CPX_NO_SOLN (cplexConstants), 60
- CPX_NODESEL_BESTBOUND (cplexConstants), 60
- CPX_NODESEL_BESTEST (cplexConstants), 60
- CPX_NODESEL_BESTEST_ALT (cplexConstants), 60
- CPX_NODESEL_DFS (cplexConstants), 60
- CPX_NONBASIC_SOLN (cplexConstants), 60
- CPX_OBJ_GAP (cplexConstants), 60
- CPX_OFF (cplexConstants), 60
- CPX_ON (cplexConstants), 60
- CPX_PARALLEL_AUTO (cplexConstants), 60
- CPX_PARALLEL_DETERMINISTIC (cplexConstants), 60
- CPX_PARALLEL_OPPORTUNISTIC (cplexConstants), 60
- CPX_PARAM_ADVIND (cplexConstants), 60
- CPX_PARAM_AGGCUTLIM (cplexConstants), 60
- CPX_PARAM_AGGFILL (cplexConstants), 60
- CPX_PARAM_AGGIND (cplexConstants), 60
- CPX_PARAM_ALL_MAX (cplexConstants), 60
- CPX_PARAM_ALL_MIN (cplexConstants), 60
- CPX_PARAM_APIENCODING (cplexConstants), 60
- CPX_PARAM_AUXROOTTHREADS (cplexConstants), 60
- CPX_PARAM_BARALG (cplexConstants), 60
- CPX_PARAM_BARCOLNZ (cplexConstants), 60
- CPX_PARAM_BARCROSSALG (cplexConstants), 60
- CPX_PARAM_BARDISPLAY (cplexConstants), 60
- CPX_PARAM_BARDSTART (cplexConstants), 60
- CPX_PARAM_BAREPCOMP (cplexConstants), 60
- CPX_PARAM_BARGROWTH (cplexConstants), 60
- CPX_PARAM_BARITLIM (cplexConstants), 60
- CPX_PARAM_BARMAXCOR (cplexConstants), 60
- CPX_PARAM_BAROBRNG (cplexConstants), 60
- CPX_PARAM_BARORDER (cplexConstants), 60
- CPX_PARAM_BARPSTART (cplexConstants), 60
- CPX_PARAM_BARQCPEPCOMP (cplexConstants), 60
- CPX_PARAM_BARSTARTALG (cplexConstants), 60
- CPX_PARAM_BASINTERVAL (cplexConstants), 60
- CPX_PARAM_BBINTERVAL (cplexConstants), 60
- CPX_PARAM_BNDSTRENIND (cplexConstants), 60
- CPX_PARAM_BRDIR (cplexConstants), 60
- CPX_PARAM_BTTOL (cplexConstants), 60
- CPX_PARAM_CALCQC PDUALS (cplexConstants), 60
- CPX_PARAM_CFILEMUL (cplexConstants), 60
- CPX_PARAM_CLIQUES (cplexConstants), 60
- CPX_PARAM_CLOCKTYPE (cplexConstants), 60
- CPX_PARAM_CLONELOG (cplexConstants), 60
- CPX_PARAM_COEREDIND (cplexConstants), 60
- CPX_PARAM_COLREADLIM (cplexConstants), 60
- CPX_PARAM_CONFLICTDISPLAY (cplexConstants), 60
- CPX_PARAM_COVERS (cplexConstants), 60
- CPX_PARAM_CRAIND (cplexConstants), 60
- CPX_PARAM_CUTLO (cplexConstants), 60
- CPX_PARAM_CUTPASS (cplexConstants), 60
- CPX_PARAM_CUTSFACOR (cplexConstants), 60

- 60
- CPX_PARAM_CUTUP (cplexConstants), 60
- CPX_PARAM_DATACHECK (cplexConstants), 60
- CPX_PARAM_DEPIND (cplexConstants), 60
- CPX_PARAM_DETTILIM (cplexConstants), 60
- CPX_PARAM_DISJCUTS (cplexConstants), 60
- CPX_PARAM_DIVETYPE (cplexConstants), 60
- CPX_PARAM_DPRIIND (cplexConstants), 60
- CPX_PARAM_EACHCUTLIM (cplexConstants), 60
- CPX_PARAM_EPAGAP (cplexConstants), 60
- CPX_PARAM_EPGAP (cplexConstants), 60
- CPX_PARAM_EPINT (cplexConstants), 60
- CPX_PARAM_EPLIN (cplexConstants), 60
- CPX_PARAM_EPMRK (cplexConstants), 60
- CPX_PARAM_EPOPT (cplexConstants), 60
- CPX_PARAM_EPOPT_H (cplexConstants), 60
- CPX_PARAM_EPPER (cplexConstants), 60
- CPX_PARAM_EPRELAX (cplexConstants), 60
- CPX_PARAM_EPRHS (cplexConstants), 60
- CPX_PARAM_EPRHS_H (cplexConstants), 60
- CPX_PARAM_FASTMIP (cplexConstants), 60
- CPX_PARAM_FEASOPTMODE (cplexConstants), 60
- CPX_PARAM_FILEENCODING (cplexConstants), 60
- CPX_PARAM_FLOWCOVERS (cplexConstants), 60
- CPX_PARAM_FLOWPATHS (cplexConstants), 60
- CPX_PARAM_FPHEUR (cplexConstants), 60
- CPX_PARAM_FRACCAND (cplexConstants), 60
- CPX_PARAM_FRACCUTS (cplexConstants), 60
- CPX_PARAM_FRACPASS (cplexConstants), 60
- CPX_PARAM_GUBCOVERS (cplexConstants), 60
- CPX_PARAM_HEURFREQ (cplexConstants), 60
- CPX_PARAM_IMPLBD (cplexConstants), 60
- CPX_PARAM_INTSOLFFILEPREFIX (cplexConstants), 60
- CPX_PARAM_INTSOLLIM (cplexConstants), 60
- CPX_PARAM_ITLIM (cplexConstants), 60
- CPX_PARAM_LANDPCUTS (cplexConstants), 60
- CPX_PARAM_LBHEUR (cplexConstants), 60
- CPX_PARAM_LPMETHOD (cplexConstants), 60
- CPX_PARAM_MFCUTS (cplexConstants), 60
- CPX_PARAM_MEMORYEMPHASIS (cplexConstants), 60
- CPX_PARAM_MIPCBREDLP (cplexConstants), 60
- CPX_PARAM_MIPDISPLAY (cplexConstants), 60
- CPX_PARAM_MIPEMPHASIS (cplexConstants), 60
- CPX_PARAM_MIPINTERVAL (cplexConstants), 60
- CPX_PARAM_MIPKAPPASTATS (cplexConstants), 60
- CPX_PARAM_MIPORDIND (cplexConstants), 60
- CPX_PARAM_MIPORDTYPE (cplexConstants), 60
- CPX_PARAM_MIPSEARCH (cplexConstants), 60
- CPX_PARAM_MIQCPSTRAT (cplexConstants), 60
- CPX_PARAM_MIRCUTS (cplexConstants), 60
- CPX_PARAM_MPSLONGNUM (cplexConstants), 60
- CPX_PARAM_NETDISPLAY (cplexConstants), 60
- CPX_PARAM_NETEPOPT (cplexConstants), 60
- CPX_PARAM_NETEPRHS (cplexConstants), 60
- CPX_PARAM_NETFIND (cplexConstants), 60
- CPX_PARAM_NETITLIM (cplexConstants), 60
- CPX_PARAM_NETPPRIIND (cplexConstants), 60
- CPX_PARAM_NODEFILEIND (cplexConstants), 60
- CPX_PARAM_NODELIM (cplexConstants), 60
- CPX_PARAM_NODESEL (cplexConstants), 60
- CPX_PARAM_NUMERICALEMPHASIS (cplexConstants), 60
- CPX_PARAM_NZREADLIM (cplexConstants), 60
- CPX_PARAM_OBJDIF (cplexConstants), 60
- CPX_PARAM_OBJLLIM (cplexConstants), 60
- CPX_PARAM_OBJULIM (cplexConstants), 60
- CPX_PARAM_PARALLELMODE (cplexConstants), 60
- CPX_PARAM_PARAMDISPLAY (cplexConstants), 60
- CPX_PARAM_PERIND (cplexConstants), 60
- CPX_PARAM_PERLIM (cplexConstants), 60
- CPX_PARAM_POLISHAFTERDETTIME (cplexConstants), 60
- CPX_PARAM_POLISHAFTEREPAGAP (cplexConstants), 60
- CPX_PARAM_POLISHAFTEREPGAP (cplexConstants), 60
- CPX_PARAM_POLISHAFTERINTSOL

- (cplexConstants), 60
- CPX_PARAM_POLISHAFTERNODE
(cplexConstants), 60
- CPX_PARAM_POLISHAFTERTIME
(cplexConstants), 60
- CPX_PARAM_POLISHTIME (cplexConstants),
60
- CPX_PARAM_POPULATELIM (cplexConstants),
60
- CPX_PARAM_PPRIIND (cplexConstants), 60
- CPX_PARAM_PREDUAL (cplexConstants), 60
- CPX_PARAM_PREIND (cplexConstants), 60
- CPX_PARAM_PRELINEAR (cplexConstants), 60
- CPX_PARAM_PREPASS (cplexConstants), 60
- CPX_PARAM_PRESLVND (cplexConstants), 60
- CPX_PARAM_PRICELIM (cplexConstants), 60
- CPX_PARAM_PROBE (cplexConstants), 60
- CPX_PARAM_PROBEDETTIME
(cplexConstants), 60
- CPX_PARAM_PROBETIME (cplexConstants), 60
- CPX_PARAM_QPMAKEPSDIND
(cplexConstants), 60
- CPX_PARAM_QPMETHOD (cplexConstants), 60
- CPX_PARAM_QPNZREADLIM (cplexConstants),
60
- CPX_PARAM_RAMPUPDETTILIM
(cplexConstants), 60
- CPX_PARAM_RAMPUPDURATION
(cplexConstants), 60
- CPX_PARAM_RAMPUPTILIM (cplexConstants),
60
- CPX_PARAM_RANDOMSEED (cplexConstants),
60
- CPX_PARAM_REDUCE (cplexConstants), 60
- CPX_PARAM_REINV (cplexConstants), 60
- CPX_PARAM_RELAXPREIND (cplexConstants),
60
- CPX_PARAM_RELOBJDIF (cplexConstants), 60
- CPX_PARAM_REPAIRTRIES (cplexConstants),
60
- CPX_PARAM_REPEATPRESOLVE
(cplexConstants), 60
- CPX_PARAM_REVERSEIND (cplexConstants),
60
- CPX_PARAM_RFILEMUL (cplexConstants), 60
- CPX_PARAM_RINSHEUR (cplexConstants), 60
- CPX_PARAM_ROWREADLIM (cplexConstants),
60
- CPX_PARAM_SCAIND (cplexConstants), 60
- CPX_PARAM_SCRIND (cplexConstants), 60
- CPX_PARAM_SIFTALG (cplexConstants), 60
- CPX_PARAM_SIFTDISPLAY (cplexConstants),
60
- CPX_PARAM_SIFTITLIM (cplexConstants), 60
- CPX_PARAM_SIMDISPLAY (cplexConstants),
60
- CPX_PARAM_SINGLIM (cplexConstants), 60
- CPX_PARAM_SINGTOL (cplexConstants), 60
- CPX_PARAM_SOLNPOOLAGAP
(cplexConstants), 60
- CPX_PARAM_SOLNPOOLCAPACITY
(cplexConstants), 60
- CPX_PARAM_SOLNPOOLGAP (cplexConstants),
60
- CPX_PARAM_SOLNPOOLINTENSITY
(cplexConstants), 60
- CPX_PARAM_SOLNPOOLREPLACE
(cplexConstants), 60
- CPX_PARAM_SOLUTIONTARGET
(cplexConstants), 60
- CPX_PARAM_STARTALG (cplexConstants), 60
- CPX_PARAM_STRONGCANDLIM
(cplexConstants), 60
- CPX_PARAM_STRONGITLIM (cplexConstants),
60
- CPX_PARAM_SUBALG (cplexConstants), 60
- CPX_PARAM_SUBMIPNODELIM
(cplexConstants), 60
- CPX_PARAM_SYMMETRY (cplexConstants), 60
- CPX_PARAM_THREADS (cplexConstants), 60
- CPX_PARAM_TILIM (cplexConstants), 60
- CPX_PARAM_TRELIM (cplexConstants), 60
- CPX_PARAM_TUNINGDETTILIM
(cplexConstants), 60
- CPX_PARAM_TUNINGDISPLAY
(cplexConstants), 60
- CPX_PARAM_TUNINGMEASURE
(cplexConstants), 60
- CPX_PARAM_TUNINGREPEAT
(cplexConstants), 60
- CPX_PARAM_TUNINGTILIM (cplexConstants),
60
- CPX_PARAM_VARSEL (cplexConstants), 60
- CPX_PARAM_WORKDIR (cplexConstants), 60
- CPX_PARAM_WORKMEM (cplexConstants), 60
- CPX_PARAM_WRITELEVEL (cplexConstants),

- 60
- CPX_PARAM_XXXIND (cplexConstants), 60
- CPX_PARAM_ZEROHALFCUTS
 - (cplexConstants), 60
- CPX_PARAMTYPE_DOUBLE (cplexConstants), 60
- CPX_PARAMTYPE_INT (cplexConstants), 60
- CPX_PARAMTYPE_LONG (cplexConstants), 60
- CPX_PARAMTYPE_NONE (cplexConstants), 60
- CPX_PARAMTYPE_STRING (cplexConstants), 60
- CPX_PPRIIND_AUTO (cplexConstants), 60
- CPX_PPRIIND_DEVEX (cplexConstants), 60
- CPX_PPRIIND_FULL (cplexConstants), 60
- CPX_PPRIIND_PARTIAL (cplexConstants), 60
- CPX_PPRIIND_STEEP (cplexConstants), 60
- CPX_PPRIIND_STEEPQSTART
 - (cplexConstants), 60
- CPX_PRECOL_AGG (cplexConstants), 60
- CPX_PRECOL_FIX (cplexConstants), 60
- CPX_PRECOL_LOW (cplexConstants), 60
- CPX_PRECOL_OTHER (cplexConstants), 60
- CPX_PRECOL_UP (cplexConstants), 60
- CPX_PREREDUCE_DUALONLY
 - (cplexConstants), 60
- CPX_PREREDUCE_NOPRIMALORDUAL
 - (cplexConstants), 60
- CPX_PREREDUCE_PRIMALANDDUAL
 - (cplexConstants), 60
- CPX_PREREDUCE_PRIMALONLY
 - (cplexConstants), 60
- CPX_PREROW_AGG (cplexConstants), 60
- CPX_PREROW_OTHER (cplexConstants), 60
- CPX_PREROW_RED (cplexConstants), 60
- CPX_PRIMAL_OBJ (cplexConstants), 60
- CPX_PRIMAL_SOLN (cplexConstants), 60
- CPX_SEMICONT (cplexConstants), 60
- CPX_SEMIINT (cplexConstants), 60
- CPX_SOLNPOOL_DIV (cplexConstants), 60
- CPX_SOLNPOOL_FIFO (cplexConstants), 60
- CPX_SOLNPOOL_FILTER_DIVERSITY
 - (cplexConstants), 60
- CPX_SOLNPOOL_FILTER_RANGE
 - (cplexConstants), 60
- CPX_SOLNPOOL_OBJ (cplexConstants), 60
- CPX_SOLUTIONTARGET_AUTO
 - (cplexConstants), 60
- CPX_SOLUTIONTARGET_FIRSTORDER
 - (cplexConstants), 60
- CPX_SOLUTIONTARGET_OPTIMALCONVEX
 - (cplexConstants), 60
- CPX_SOLUTIONTARGET_OPTIMALGLOBAL
 - (cplexConstants), 60
- CPX_STAT_ABORT_DETTIME_LIM
 - (cplexConstants), 60
- CPX_STAT_ABORT_DUAL_OBJ_LIM
 - (cplexConstants), 60
- CPX_STAT_ABORT_IT_LIM (cplexConstants), 60
- CPX_STAT_ABORT_OBJ_LIM
 - (cplexConstants), 60
- CPX_STAT_ABORT_PRIM_OBJ_LIM
 - (cplexConstants), 60
- CPX_STAT_ABORT_TIME_LIM
 - (cplexConstants), 60
- CPX_STAT_ABORT_USER (cplexConstants), 60
- CPX_STAT_CONFLICT_ABORT_CONTRADICTION
 - (cplexConstants), 60
- CPX_STAT_CONFLICT_ABORT_DETTIME_LIM
 - (cplexConstants), 60
- CPX_STAT_CONFLICT_ABORT_IT_LIM
 - (cplexConstants), 60
- CPX_STAT_CONFLICT_ABORT_MEM_LIM
 - (cplexConstants), 60
- CPX_STAT_CONFLICT_ABORT_NODE_LIM
 - (cplexConstants), 60
- CPX_STAT_CONFLICT_ABORT_OBJ_LIM
 - (cplexConstants), 60
- CPX_STAT_CONFLICT_ABORT_TIME_LIM
 - (cplexConstants), 60
- CPX_STAT_CONFLICT_ABORT_USER
 - (cplexConstants), 60
- CPX_STAT_CONFLICT_FEASIBLE
 - (cplexConstants), 60
- CPX_STAT_CONFLICT_MINIMAL
 - (cplexConstants), 60
- CPX_STAT_FEASIBLE (cplexConstants), 60
- CPX_STAT_FEASIBLE_RELAXED_INF
 - (cplexConstants), 60
- CPX_STAT_FEASIBLE_RELAXED_QUAD
 - (cplexConstants), 60
- CPX_STAT_FEASIBLE_RELAXED_SUM
 - (cplexConstants), 60
- CPX_STAT_FIRSTORDER (cplexConstants), 60
- CPX_STAT_INFEASIBLE (cplexConstants), 60
- CPX_STAT_INFForUNBD (cplexConstants), 60

- CPX_STAT_NUM_BEST (cplexConstants), 60
 CPX_STAT_OPTIMAL (cplexConstants), 60
 CPX_STAT_OPTIMAL_FACE_UNBOUNDED
 (cplexConstants), 60
 CPX_STAT_OPTIMAL_INFEAS
 (cplexConstants), 60
 CPX_STAT_OPTIMAL_RELAXED_INF
 (cplexConstants), 60
 CPX_STAT_OPTIMAL_RELAXED_QUAD
 (cplexConstants), 60
 CPX_STAT_OPTIMAL_RELAXED_SUM
 (cplexConstants), 60
 CPX_STAT_UNBOUNDED (cplexConstants), 60
 CPX_STR_PARAM_MAX (cplexConstants), 60
 CPX_SUM_COMP_SLACK (cplexConstants), 60
 CPX_SUM_DUAL_INFEAS (cplexConstants), 60
 CPX_SUM_DUAL_RESIDUAL (cplexConstants),
 60
 CPX_SUM_INDSLACK_INFEAS
 (cplexConstants), 60
 CPX_SUM_INT_INFEAS (cplexConstants), 60
 CPX_SUM_PI (cplexConstants), 60
 CPX_SUM_PRIMAL_INFEAS (cplexConstants),
 60
 CPX_SUM_PRIMAL_RESIDUAL
 (cplexConstants), 60
 CPX_SUM_PWSLACK_INFEAS
 (cplexConstants), 60
 CPX_SUM_QCPRIMAL_RESIDUAL
 (cplexConstants), 60
 CPX_SUM_QCSLACK (cplexConstants), 60
 CPX_SUM_QCSLACK_INFEAS
 (cplexConstants), 60
 CPX_SUM_RED_COST (cplexConstants), 60
 CPX_SUM_SCALED_DUAL_INFEAS
 (cplexConstants), 60
 CPX_SUM_SCALED_DUAL_RESIDUAL
 (cplexConstants), 60
 CPX_SUM_SCALED_PI (cplexConstants), 60
 CPX_SUM_SCALED_PRIMAL_INFEAS
 (cplexConstants), 60
 CPX_SUM_SCALED_PRIMAL_RESIDUAL
 (cplexConstants), 60
 CPX_SUM_SCALED_RED_COST
 (cplexConstants), 60
 CPX_SUM_SCALED_SLACK (cplexConstants),
 60
 CPX_SUM_SCALED_X (cplexConstants), 60
 CPX_SUM_SLACK (cplexConstants), 60
 CPX_SUM_X (cplexConstants), 60
 CPX_TUNE_ABORT (cplexConstants), 60
 CPX_TUNE_AVERAGE (cplexConstants), 60
 CPX_TUNE_DETTILIM (cplexConstants), 60
 CPX_TUNE_MINMAX (cplexConstants), 60
 CPX_TUNE_TILIM (cplexConstants), 60
 CPX_TYPE_ANY (cplexConstants), 60
 CPX_TYPE_SOS1 (cplexConstants), 60
 CPX_TYPE_SOS2 (cplexConstants), 60
 CPX_TYPE_USER (cplexConstants), 60
 CPX_TYPE_VAR (cplexConstants), 60
 CPX_USECUT_FILTER (cplexConstants), 60
 CPX_USECUT_FORCE (cplexConstants), 60
 CPX_USECUT_PURGE (cplexConstants), 60
 CPX_VARSEL_DEFAULT (cplexConstants), 60
 CPX_VARSEL_MAXINFEAS (cplexConstants),
 60
 CPX_VARSEL_MININFEAS (cplexConstants),
 60
 CPX_VARSEL_PSEUDO (cplexConstants), 60
 CPX_VARSEL_PSEUDOREduced
 (cplexConstants), 60
 CPX_VARSEL_STRONG (cplexConstants), 60
 CPX_WRITELEVEL_ALLVARS
 (cplexConstants), 60
 CPX_WRITELEVEL_AUTO (cplexConstants), 60
 CPX_WRITELEVEL_DISCRETEVARS
 (cplexConstants), 60
 CPX_WRITELEVEL_NONZERODISCRETEVARS
 (cplexConstants), 60
 CPX_WRITELEVEL_NONZEROVARS
 (cplexConstants), 60
 CPXaddcols (addColsCPLEX), 7
 CPXaddfpdest (addFpDestCPLEX), 9
 CPXaddindconstr (addIndConstrCPLEX), 10
 CPXaddmipstarts (addMIPstartsCPLEX), 11
 CPXaddqconstr (addQConstrCPLEX), 12
 CPXaddrows (addRowsCPLEX), 13
 CPXbaropt (baroptCPLEX), 14
 CPXbasicpresolve (basicPresolveCPLEX),
 16
 CPXboundsa (boundSaCPLEX), 17
 CPXcheckaddcols (checkAddColsCPLEX), 18
 CPXcheckaddrows (checkAddRowsCPLEX), 19
 CPXcheckchgcoeflist
 (checkChgCoefListCPLEX), 20
 CPXcheckcopyctype

- (checkCopyColTypeCPLEX), 21
- CPXcheckcopylp (checkCopyLpCPLEX), 22
- CPXcheckcopylpwnames
 - (checkCopyLpwNamesCPLEX), 23
- CPXcheckcopyqpsep
 - (checkCopyQPsepCPLEX), 25
- CPXcheckcopyquad (checkCopyQuadCPLEX), 26
- CPXcheckvals (checkValsCPLEX), 27
- CPXchgbds (chgBndsCPLEX), 28
- CPXchgcoef (chgCoefCPLEX), 29
- CPXchgcoeflist (chgCoefListCPLEX), 30
- CPXchgcolname (chgColNameCPLEX), 31
- CPXchgctype (chgColTypeCPLEX), 33
- CPXchgmipstarts (chgMIPstartsCPLEX), 34
- CPXchgname (chgNameCPLEX), 35
- CPXchgobj (chgObjCPLEX), 36
- CPXchgobjsen (setObjDirCPLEX), 219
- CPXchgprobname (chgProbNameCPLEX), 37
- CPXchgprobtype (chgProbTypeCPLEX), 38
- CPXchgqpcoef (chgQPcoefCPLEX), 39
- CPXchgrhs (chgRhsCPLEX), 40
- CPXchgrngval (chgRngValCPLEX), 41
- CPXchgrowname (chgRowNameCPLEX), 42
- CPXchgsense (chgSenseCPLEX), 43
- CPXcleanup (cleanupCoefCPLEX), 44
- CPXcloneprob (cloneProbCPLEX), 45
- CPXcloseCPLEX (closeEnvCPLEX), 46
- CPXclpwrite (clpWriteCPLEX), 49
- CPXcompletelp (completelpCPLEX), 50
- CPXcopybase (copyBaseCPLEX), 51
- CPXcopyctype (copyColTypeCPLEX), 52
- CPXcopylp (copyLpCPLEX), 53
- CPXcopylpwnames (copyLpwNamesCPLEX), 54
- CPXcopyobjname (copyObjNameCPLEX), 55
- CPXcopyorder (copyOrderCPLEX), 56
- CPXcopyqpsep (copyQPsepCPLEX), 57
- CPXcopyquad (copyQuadCPLEX), 58
- CPXcopystart (copyStartCPLEX), 59
- CPXcreateprob (initProbCPLEX), 187
- CPXdelcols (delColsCPLEX), 84
- CPXdelfpdest (delFpDestCPLEX), 85
- CPXdelindconstrs (delIndConstrsCPLEX), 86
- CPXdelmipstarts (delMIPstartsCPLEX), 87
- CPXdelnames (delNamesCPLEX), 88
- CPXdelqconstrs (delQConstrsCPLEX), 90
- CPXdelrows (delRowsCPLEX), 91
- CPXdelsetcols (delSetColsCPLEX), 92
- CPXdelsetrows (delSetRowsCPLEX), 93
- CPXdisconnectchannel
 - (disconnectChannelCPLEX), 95
- CPXdualopt (dualoptCPLEX), 96
- CPXdualwrite (dualWriteCPLEX), 97
- CPXERR_BAD_MULTIOBJ_ATTR
 - (cplexConstants), 60
- CPXERR_CALLBACK_INCONSISTENT
 - (cplexConstants), 60
- CPXERR_CAND_NOT_POINT (cplexConstants), 60
- CPXERR_CAND_NOT_RAY (cplexConstants), 60
- CPXERR_MULTIOBJ_SUBPROB_SOLVE
 - (cplexConstants), 60
- CPXERR_NEGATIVE_SURPLUS
 - (cplexConstants), 60
- CPXERR_NO_OBJ_NAME (cplexConstants), 60
- CPXERR_NO_SENSIT (cplexConstants), 60
- CPXERR_NOT_FOR_MULTIOBJ
 - (cplexConstants), 60
- CPXfclose (closeFileCPLEX), 47
- CPXfeasopt (feasOptCPLEX), 98
- CPXflushchannel (flushChannelCPLEX), 100
- CPXflushstdchannels
 - (flushStdChannelsCPLEX), 101
- CPXfopen (openFileCPLEX), 194
- CPXfputs (fileputCPLEX), 99
- CPXfreepresolve (freePresolveCPLEX), 102
- CPXfreeprob (delProbCPLEX), 89
- CPXgetbase (getBaseCPLEX), 103
- CPXgetbestobjval (getBestObjValCPLEX), 104
- CPXgetchannels (getChannelsCPLEX), 105
- CPXgetchgparam (getChgParmCPLEX), 106
- CPXgetcoef (getCoefCPLEX), 107
- CPXgetcolindex (getColIndexCPLEX), 108
- CPXgetcolinfeas (getColInfeasCPLEX), 109
- CPXgetcolname (getColNameCPLEX), 110
- CPXgetcols (getColsCPLEX), 111
- CPXgetconflict (getConflictCPLEX), 113
- CPXgetconflictext
 - (getConflictExtCPLEX), 114
- CPXgetctype (getColTypeCPLEX), 112
- CPXgetcutoff (getCutoffCPLEX), 115
- CPXgetdblparam (getDb1ParmCPLEX), 116
- CPXgetdblquality (getDb1QualCPLEX), 117
- CPXgetdj (getDjCPLEX), 119

- CPXgetdsbcnt (getDbCntCPLEX), 118
- CPXgeterrorstring (getErrorStrCPLEX), 120
- CPXgetgrad (getGradCPLEX), 121
- CPXgetindconstr (getIndConstrCPLEX), 122
- CPXgetintparam (getIntParmCPLEX), 127
- CPXgetintquality (getIntQualCPLEX), 128
- CPXgetitcnt (getItCntCPLEX), 129
- CPXgetlb (getLowerBndsCPLEX), 134
- CPXgetlogfile (getLogFileCPLEX), 130
- CPXgetlogfilename (getLogFileNameCPLEX), 131
- CPXgetlongparam (getLongParmCPLEX), 132
- CPXgetmethod (getMethodCPLEX), 135
- CPXgetmiprelgap (getMIPrelGapCPLEX), 136
- CPXgetmipstartindex (getMIPstartIndexCPLEX), 137
- CPXgetmipstartname (getMIPstartNameCPLEX), 138
- CPXgetmipstarts (getMIPstartsCPLEX), 139
- CPXgetnumcols (getNumColsCPLEX), 140
- CPXgetnummipstarts (getNumMIPstartsCPLEX), 141
- CPXgetnumnz (getNumNnzCPLEX), 142
- CPXgetnumqconstrs (getNumQConstrsCPLEX), 143
- CPXgetnumqpz (getNumQPnzCPLEX), 144
- CPXgetnumquad (getNumQuadCPLEX), 145
- CPXgetnumrows (getNumRowsCPLEX), 146
- CPXgetobj (getObjCPLEX), 147
- CPXgetobjname (getObjNameCPLEX), 149
- CPXgetobjoffset (getObjOffsetCPLEX), 150
- CPXgetobjsen (getObjDirCPLEX), 148
- CPXgetobjval (getObjValCPLEX), 151
- CPXgetorder (getOrderCPLEX), 152
- CPXgetparamhiername (getParamHierNameCPLEX), 153
- CPXgetparamname (getParamNameCPLEX), 154
- CPXgetparamnum (getParamNumCPLEX), 155
- CPXgetparamtype (getParamTypeCPLEX), 156
- CPXgetphase1cnt (getPhase1CntCPLEX), 157
- CPXgetpi (getPiCPLEX), 158
- CPXgetprestat (getPreStatCPLEX), 159
- CPXgetprobname (getProbNameCPLEX), 160
- CPXgetprobtype (getProbTypeCPLEX), 161
- CPXgetqconstr (getQConstrCPLEX), 163
- CPXgetqpcoef (getQPcoefCPLEX), 164
- CPXgetquad (getQuadCPLEX), 165
- CPXgetredlp (getRedLpCPLEX), 166
- CPXgetrhs (getRhsCPLEX), 167
- CPXgetrngval (getRngValCPLEX), 168
- CPXgetrowindex (getRowIndexCPLEX), 169
- CPXgetrowinfeas (getRowInfeasCPLEX), 170
- CPXgetrowname (getRowNameCPLEX), 171
- CPXgetrows (getRowsCPLEX), 172
- CPXgetsense (getSenseCPLEX), 173
- CPXgetsiftitcnt (getSiftItCntCPLEX), 174
- CPXgetsiftphase1cnt (getSiftPhase1CntCPLEX), 175
- CPXgetslack (getSlackCPLEX), 176
- CPXgetstat (getStatCPLEX), 177
- CPXgetstatstring (getStatStrCPLEX), 178
- CPXgetstrparam (getStrParmCPLEX), 179
- CPXgetsubmethod (getSubMethodCPLEX), 180
- CPXgetsubstat (getSubStatCPLEX), 181
- CPXgettime (getTimeCPLEX), 182
- CPXgetub (getUpperBndsCPLEX), 184
- CPXgetx (getProbVarCPLEX), 162
- CPXhybbaropt (hybbaroptCPLEX), 185
- CPXhybnetopt (hybnetoptCPLEX), 186
- CPXinfodblparam (getInfoDblParmCPLEX), 123
- CPXinfointparam (getInfoIntParmCPLEX), 124
- CPXinfoolongparam (getInfoLongParmCPLEX), 125
- CPXinfostrparam (getInfoStrParmCPLEX), 126
- CPXlpopt (lpoptCPLEX), 188
- CPXmbasewrite (baseWriteCPLEX), 15
- CPXMI_SAMECOEFF_COL (cplexConstants), 60
- CPXMI_SAMECOEFF_IND (cplexConstants), 60
- CPXMI_SAMECOEFF_LAZY (cplexConstants), 60
- CPXMI_SAMECOEFF_OBJ (cplexConstants), 60
- CPXMI_SAMECOEFF_QLIN (cplexConstants), 60
- CPXMI_SAMECOEFF_QUAD (cplexConstants), 60
- CPXMI_SAMECOEFF_RHS (cplexConstants), 60
- CPXMI_SAMECOEFF_ROW (cplexConstants), 60
- CPXMI_SAMECOEFF_UCUT (cplexConstants), 60
- CPXMIP_ABORT_FEAS (cplexConstants), 60
- CPXMIP_ABORT_INFEAS (cplexConstants), 60
- CPXMIP_ABORT_RELAXED (cplexConstants),

- 60
- CPXMIP_DETTIME_LIM_FEAS
(cplexConstants), 60
- CPXMIP_DETTIME_LIM_INFEAS
(cplexConstants), 60
- CPXMIP_FAIL_FEAS (cplexConstants), 60
- CPXMIP_FAIL_FEAS_NO_TREE
(cplexConstants), 60
- CPXMIP_FAIL_INFEAS (cplexConstants), 60
- CPXMIP_FAIL_INFEAS_NO_TREE
(cplexConstants), 60
- CPXMIP_FEASIBLE (cplexConstants), 60
- CPXMIP_FEASIBLE_RELAXED_INF
(cplexConstants), 60
- CPXMIP_FEASIBLE_RELAXED_QUAD
(cplexConstants), 60
- CPXMIP_FEASIBLE_RELAXED_SUM
(cplexConstants), 60
- CPXMIP_INFEASIBLE (cplexConstants), 60
- CPXMIP_INFForUNBD (cplexConstants), 60
- CPXMIP_MEM_LIM_FEAS (cplexConstants), 60
- CPXMIP_MEM_LIM_INFEAS (cplexConstants),
60
- CPXMIP_NODE_LIM_FEAS (cplexConstants),
60
- CPXMIP_NODE_LIM_INFEAS
(cplexConstants), 60
- CPXMIP_OPTIMAL (cplexConstants), 60
- CPXMIP_OPTIMAL_INFEAS (cplexConstants),
60
- CPXMIP_OPTIMAL_POPULATED
(cplexConstants), 60
- CPXMIP_OPTIMAL_POPULATED_TOL
(cplexConstants), 60
- CPXMIP_OPTIMAL_RELAXED_INF
(cplexConstants), 60
- CPXMIP_OPTIMAL_RELAXED_QUAD
(cplexConstants), 60
- CPXMIP_OPTIMAL_RELAXED_SUM
(cplexConstants), 60
- CPXMIP_OPTIMAL_TOL (cplexConstants), 60
- CPXMIP_POPULATESOL_LIM
(cplexConstants), 60
- CPXMIP_SOL_LIM (cplexConstants), 60
- CPXMIP_TIME_LIM_FEAS (cplexConstants),
60
- CPXMIP_TIME_LIM_INFEAS
(cplexConstants), 60
- CPXMIP_UNBOUNDED (cplexConstants), 60
- CPXmipopt (mipoptCplex), 189
- CPXNET_NO_DISPLAY_OBJECTIVE
(cplexConstants), 60
- CPXNET_PENALIZED_OBJECTIVE
(cplexConstants), 60
- CPXNET_PRICE_AUTO (cplexConstants), 60
- CPXNET_PRICE_MULT_PART
(cplexConstants), 60
- CPXNET_PRICE_PARTIAL (cplexConstants),
60
- CPXNET_PRICE_SORT_MULT_PART
(cplexConstants), 60
- CPXNET_TRUE_OBJECTIVE (cplexConstants),
60
- CPXnewcols (newColsCplex), 190
- CPXnewrows (newRowsCplex), 191
- CPXobjsa (objSaCplex), 192
- CPXopenCplex (openEnvCplex), 193
- CPXordwrite (ordWriteCplex), 196
- CPXPARAM_Advance (cplexConstants), 60
- CPXPARAM_Barrier_Algorithm
(cplexConstants), 60
- CPXPARAM_Barrier_ColNonzeros
(cplexConstants), 60
- CPXPARAM_Barrier_ConvergeTol
(cplexConstants), 60
- CPXPARAM_Barrier_Crossover
(cplexConstants), 60
- CPXPARAM_Barrier_Display
(cplexConstants), 60
- CPXPARAM_Barrier_Limits_Corrections
(cplexConstants), 60
- CPXPARAM_Barrier_Limits_Growth
(cplexConstants), 60
- CPXPARAM_Barrier_Limits_Iteration
(cplexConstants), 60
- CPXPARAM_Barrier_Limits_ObjRange
(cplexConstants), 60
- CPXPARAM_Barrier_Ordering
(cplexConstants), 60
- CPXPARAM_Barrier_QCConvergeTol
(cplexConstants), 60
- CPXPARAM_Barrier_StartAlg
(cplexConstants), 60
- CPXPARAM_ClockType (cplexConstants), 60
- CPXPARAM_Conflict_Display
(cplexConstants), 60

- CPXPARAM_DetTimeLimit (cplexConstants),
60
- CPXPARAM_DistMIP_Rampup_DetTimeLimit
(cplexConstants), 60
- CPXPARAM_DistMIP_Rampup_Duration
(cplexConstants), 60
- CPXPARAM_DistMIP_Rampup_TimeLimit
(cplexConstants), 60
- CPXPARAM_Emphasis_Memory
(cplexConstants), 60
- CPXPARAM_Emphasis_MIP (cplexConstants),
60
- CPXPARAM_Emphasis_Numerical
(cplexConstants), 60
- CPXPARAM_Feasopt_Mode (cplexConstants),
60
- CPXPARAM_Feasopt_Tolerance
(cplexConstants), 60
- CPXPARAM_LPMethod (cplexConstants), 60
- CPXPARAM_MIP_Cuts_Cliques
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_Covers
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_Disjunctive
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_FlowCovers
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_Gomory
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_GUBCovers
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_Implied
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_LiftProj
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_MCFCut
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_MIRCut
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_PathCut
(cplexConstants), 60
- CPXPARAM_MIP_Cuts_ZeroHalfCut
(cplexConstants), 60
- CPXPARAM_MIP_Display (cplexConstants),
60
- CPXPARAM_MIP_Interval (cplexConstants),
60
- CPXPARAM_MIP_Limits_AggForCut
(cplexConstants), 60
- CPXPARAM_MIP_Limits_AuxRootThreads
(cplexConstants), 60
- CPXPARAM_MIP_Limits_CutPasses
(cplexConstants), 60
- CPXPARAM_MIP_Limits_CutsFactor
(cplexConstants), 60
- CPXPARAM_MIP_Limits_EachCutLimit
(cplexConstants), 60
- CPXPARAM_MIP_Limits_GomoryCand
(cplexConstants), 60
- CPXPARAM_MIP_Limits_GomoryPass
(cplexConstants), 60
- CPXPARAM_MIP_Limits_Nodes
(cplexConstants), 60
- CPXPARAM_MIP_Limits_PolishTime
(cplexConstants), 60
- CPXPARAM_MIP_Limits_Populate
(cplexConstants), 60
- CPXPARAM_MIP_Limits_ProbeDetTime
(cplexConstants), 60
- CPXPARAM_MIP_Limits_ProbeTime
(cplexConstants), 60
- CPXPARAM_MIP_Limits_RepairTries
(cplexConstants), 60
- CPXPARAM_MIP_Limits_Solutions
(cplexConstants), 60
- CPXPARAM_MIP_Limits_StrongCand
(cplexConstants), 60
- CPXPARAM_MIP_Limits_StrongIt
(cplexConstants), 60
- CPXPARAM_MIP_Limits_SubMIPNodeLim
(cplexConstants), 60
- CPXPARAM_MIP_Limits_TreeMemory
(cplexConstants), 60
- CPXPARAM_MIP_OrderType
(cplexConstants), 60
- CPXPARAM_MIP_PolishAfter_AbsMIPGap
(cplexConstants), 60
- CPXPARAM_MIP_PolishAfter_DetTime
(cplexConstants), 60
- CPXPARAM_MIP_PolishAfter_MIPGap
(cplexConstants), 60
- CPXPARAM_MIP_PolishAfter_Nodes
(cplexConstants), 60
- CPXPARAM_MIP_PolishAfter_Solutions
(cplexConstants), 60
- CPXPARAM_MIP_PolishAfter_Time

- (cplexConstants), 60
- CPXPARAM_MIP_Pool_AbsGap (cplexConstants), 60
- CPXPARAM_MIP_Pool_Capacity (cplexConstants), 60
- CPXPARAM_MIP_Pool_Intensity (cplexConstants), 60
- CPXPARAM_MIP_Pool_RelGap (cplexConstants), 60
- CPXPARAM_MIP_Pool_Replace (cplexConstants), 60
- CPXPARAM_MIP_Strategy_Backtrack (cplexConstants), 60
- CPXPARAM_MIP_Strategy_BBInterval (cplexConstants), 60
- CPXPARAM_MIP_Strategy_Branch (cplexConstants), 60
- CPXPARAM_MIP_Strategy_CallbackReducedLP (cplexConstants), 60
- CPXPARAM_MIP_Strategy_Dive (cplexConstants), 60
- CPXPARAM_MIP_Strategy_File (cplexConstants), 60
- CPXPARAM_MIP_Strategy_FPHeur (cplexConstants), 60
- CPXPARAM_MIP_Strategy_HeuristicFreq (cplexConstants), 60
- CPXPARAM_MIP_Strategy_KappaStats (cplexConstants), 60
- CPXPARAM_MIP_Strategy_LBHeur (cplexConstants), 60
- CPXPARAM_MIP_Strategy_MIQCPStrat (cplexConstants), 60
- CPXPARAM_MIP_Strategy_NodeSelect (cplexConstants), 60
- CPXPARAM_MIP_Strategy_Order (cplexConstants), 60
- CPXPARAM_MIP_Strategy_PresolveNode (cplexConstants), 60
- CPXPARAM_MIP_Strategy_Probe (cplexConstants), 60
- CPXPARAM_MIP_Strategy_RINSHeur (cplexConstants), 60
- CPXPARAM_MIP_Strategy_Search (cplexConstants), 60
- CPXPARAM_MIP_Strategy_StartAlgorithm (cplexConstants), 60
- CPXPARAM_MIP_Strategy_SubAlgorithm (cplexConstants), 60
- CPXPARAM_MIP_Strategy_SubMIPScale (cplexConstants), 60
- CPXPARAM_MIP_Strategy_SubMIPStartAlg (cplexConstants), 60
- CPXPARAM_MIP_Strategy_SubMIPSubAlg (cplexConstants), 60
- CPXPARAM_MIP_Strategy_VariableSelect (cplexConstants), 60
- CPXPARAM_MIP_Tolerances_AbsMIPGap (cplexConstants), 60
- CPXPARAM_MIP_Tolerances_Integrality (cplexConstants), 60
- CPXPARAM_MIP_Tolerances_LowerCutoff (cplexConstants), 60
- CPXPARAM_MIP_Tolerances_MIPGap (cplexConstants), 60
- CPXPARAM_MIP_Tolerances_ObjDifference (cplexConstants), 60
- CPXPARAM_MIP_Tolerances_RelObjDifference (cplexConstants), 60
- CPXPARAM_MIP_Tolerances_UpperCutoff (cplexConstants), 60
- CPXPARAM_MultiObjective_Display (cplexConstants), 60
- CPXPARAM_Network_Display (cplexConstants), 60
- CPXPARAM_Network_Iterations (cplexConstants), 60
- CPXPARAM_Network_NetFind (cplexConstants), 60
- CPXPARAM_Network_Pricing (cplexConstants), 60
- CPXPARAM_Network_Tolerances_Feasibility (cplexConstants), 60
- CPXPARAM_Network_Tolerances_Optimality (cplexConstants), 60
- CPXPARAM_Output_CloneLog (cplexConstants), 60
- CPXPARAM_Output_IntSolFilePrefix (cplexConstants), 60
- CPXPARAM_Output_MPSPong (cplexConstants), 60
- CPXPARAM_Output_WriteLevel (cplexConstants), 60
- CPXPARAM_Parallel (cplexConstants), 60
- CPXPARAM_ParamDisplay (cplexConstants), 60

- CPXPARAM_Preprocessing_Aggregator
(cplexConstants), 60
- CPXPARAM_Preprocessing_BoundStrength
(cplexConstants), 60
- CPXPARAM_Preprocessing_CoeffReduce
(cplexConstants), 60
- CPXPARAM_Preprocessing_Dependency
(cplexConstants), 60
- CPXPARAM_Preprocessing_Dual
(cplexConstants), 60
- CPXPARAM_Preprocessing_Fill
(cplexConstants), 60
- CPXPARAM_Preprocessing_Folding
(cplexConstants), 60
- CPXPARAM_Preprocessing_Linear
(cplexConstants), 60
- CPXPARAM_Preprocessing_NumPass
(cplexConstants), 60
- CPXPARAM_Preprocessing_Presolve
(cplexConstants), 60
- CPXPARAM_Preprocessing_QCPDuals
(cplexConstants), 60
- CPXPARAM_Preprocessing_QPMakePSD
(cplexConstants), 60
- CPXPARAM_Preprocessing_Reduce
(cplexConstants), 60
- CPXPARAM_Preprocessing_Relax
(cplexConstants), 60
- CPXPARAM_Preprocessing_RepeatPresolve
(cplexConstants), 60
- CPXPARAM_Preprocessing_Symmetry
(cplexConstants), 60
- CPXPARAM_QPMethod (cplexConstants), 60
- CPXPARAM_RandomSeed (cplexConstants), 60
- CPXPARAM_Read_APIEncoding
(cplexConstants), 60
- CPXPARAM_Read_Constraints
(cplexConstants), 60
- CPXPARAM_Read_DataCheck
(cplexConstants), 60
- CPXPARAM_Read_FileEncoding
(cplexConstants), 60
- CPXPARAM_Read_Nonzeros
(cplexConstants), 60
- CPXPARAM_Read_QPNonzeros
(cplexConstants), 60
- CPXPARAM_Read_Scale (cplexConstants), 60
- CPXPARAM_Read_Variables
(cplexConstants), 60
- CPXPARAM_Read_WarningLimit
(cplexConstants), 60
- CPXPARAM_Record (cplexConstants), 60
- CPXPARAM_ScreenOutput (cplexConstants),
60
- CPXPARAM_Sifting_Algorithm
(cplexConstants), 60
- CPXPARAM_Sifting_Display
(cplexConstants), 60
- CPXPARAM_Sifting_Iterations
(cplexConstants), 60
- CPXPARAM_Simplex_Crash
(cplexConstants), 60
- CPXPARAM_Simplex_DGradient
(cplexConstants), 60
- CPXPARAM_Simplex_Display
(cplexConstants), 60
- CPXPARAM_Simplex_Limits_Iterations
(cplexConstants), 60
- CPXPARAM_Simplex_Limits_LowerObj
(cplexConstants), 60
- CPXPARAM_Simplex_Limits_Perturbation
(cplexConstants), 60
- CPXPARAM_Simplex_Limits_Singularity
(cplexConstants), 60
- CPXPARAM_Simplex_Limits_UpperObj
(cplexConstants), 60
- CPXPARAM_Simplex_Perturbation_Constant
(cplexConstants), 60
- CPXPARAM_Simplex_Perturbation_Indicator
(cplexConstants), 60
- CPXPARAM_Simplex_PGradient
(cplexConstants), 60
- CPXPARAM_Simplex_Pricing
(cplexConstants), 60
- CPXPARAM_Simplex_Refactor
(cplexConstants), 60
- CPXPARAM_Simplex_Tolerances_Feasibility
(cplexConstants), 60
- CPXPARAM_Simplex_Tolerances_Markowitz
(cplexConstants), 60
- CPXPARAM_Simplex_Tolerances_Optimality
(cplexConstants), 60
- CPXPARAM_SolutionTarget
(cplexConstants), 60
- CPXPARAM_Threads (cplexConstants), 60
- CPXPARAM_TimeLimit (cplexConstants), 60

- CPXPARAM_Tune_DetTimeLimit
(cplexConstants), 60
- CPXPARAM_Tune_Display (cplexConstants),
60
- CPXPARAM_Tune_Measure (cplexConstants),
60
- CPXPARAM_Tune_Repeat (cplexConstants),
60
- CPXPARAM_Tune_TimeLimit
(cplexConstants), 60
- CPXPARAM_WorkDir (cplexConstants), 60
- CPXPARAM_WorkMem (cplexConstants), 60
- CPXpreslvwrite (preslvWriteCplex), 197
- CPXpresolve (presolveCplex), 198
- CPXprimopt (primoptCplex), 199
- CPXPROB_FIXEDMILP (cplexConstants), 60
- CPXPROB_FIXEDMIQP (cplexConstants), 60
- CPXPROB_LP (cplexConstants), 60
- CPXPROB_MILP (cplexConstants), 60
- CPXPROB_MIQCP (cplexConstants), 60
- CPXPROB_MIQP (cplexConstants), 60
- CPXPROB_NODEL (cplexConstants), 60
- CPXPROB_NODEQCP (cplexConstants), 60
- CPXPROB_NODEQP (cplexConstants), 60
- CPXPROB_QCP (cplexConstants), 60
- CPXPROB_QP (cplexConstants), 60
- CPXqpopt (qpoptCplex), 200
- CPXreadcopybase (readCopyBaseCplex), 201
- CPXreadcopymipstarts
(readCopyMIPstartsCplex), 202
- CPXreadcopyorder (readCopyOrderCplex),
203
- CPXreadcopyparam (readCopyParmCplex),
204
- CPXreadcopyprob (readCopyProbCplex), 205
- CPXreadcopysol (readCopySolCplex), 206
- CPXrefineconflict
(refineConflictCplex), 207
- CPXrefineconflicttext
(refineConflictExtCplex), 208
- CPXrefinemipstartconflict
(refineMIPstartConflictCplex),
209
- CPXrefinemipstartconflicttext
(refineMIPstartConflictExtCplex),
210
- CPXrhssa (rhsSaCplex), 212
- CPXsetdblparam (setDb1ParmCplex), 213
- CPXsetdefaults (setDefaultParmCplex),
214
- CPXsetintparam (setIntParmCplex), 215
- CPXsetlogfile (setLogFileCplex), 216
- CPXsetlogfilename
(setLogFileNameCplex), 217
- CPXsetlongparam (setLongParmCplex), 218
- CPXsetstrparam (setStrParmCplex), 220
- CPXsetterminate (delTerminateCplex), 94
- CPXsiftopt (siftoptCplex), 222
- CPXsolninfo (solnInfoCplex), 223
- CPXsolution (solutionCplex), 224
- CPXsolwrite (solWriteCplex), 225
- CPXtightenbds (tightenBndsCplex), 227
- CPXtuneparam (tuneParmCplex), 228
- CPXunscaleprob (unscaleProbCplex), 229
- CPXversion (getVersionCplex), 185
- CPXwritemipstarts
(writeMIPstartsCplex), 230
- CPXwriteparam (writeParmCplex), 231
- CPXwriteprob (writeProbCplex), 232
- delColsCplex, 84
- delFpDestCplex, 9, 85
- delIndConstrsCplex, 86
- delMIPstartsCplex, 87
- delNamesCplex, 88
- delProbCplex, 89, 188
- delQConstrsCplex, 90
- delRowsCplex, 91
- delSetColsCplex, 92
- delSetRowsCplex, 93
- delTerminateCplex, 44, 94, 200, 221
- disconnectChannelCplex, 95, 100, 101, 105
- dualoptCplex, 96
- dualwriteCplex, 97
- err (cplexError-class), 82
- err, cplexError-method
(cplexError-class), 82
- errmsg (cplexError-class), 82
- errmsg, cplexError-method
(cplexError-class), 82
- errnum (cplexError-class), 82
- errnum, cplexError-method
(cplexError-class), 82
- errnum<- (cplexError-class), 82
- errnum<-, cplexError-method
(cplexError-class), 82

- feasOptCPLEX, 98
- fileputCPLEX, 47, 99, 182, 194
- flushChannelCPLEX, 95, 100, 101, 105
- flushStdChannelsCPLEX, 95, 100, 101, 105
- freePresolveCPLEX, 102

- getBaseCPLEX, 103
- getBestObjValCPLEX, 104, 136
- getChannelsCPLEX, 95, 100, 101, 105
- getChgParmCPLEX, 106, 157
- getCoefCPLEX, 107
- getColIndexCPLEX, 108
- getColInfeasCPLEX, 98, 109
- getColNameCPLEX, 110
- getColsCPLEX, 111
- getColTypeCPLEX, 112
- getConflictCPLEX, 113, 207, 209
- getConflictExtCPLEX, 114
- getCutoffCPLEX, 115
- getDb1ParmCPLEX, 116
- getDb1QualCPLEX, 117
- getDb3CntCPLEX, 118
- getDjCPLEX, 119
- getErrorStrCPLEX, 120, 178
- getGradCPLEX, 121
- getIndConstrCPLEX, 122
- getInfoDb1ParmCPLEX, 123
- getInfoIntParmCPLEX, 124, 126
- getInfoLongParmCPLEX, 125
- getInfoStrParmCPLEX, 126
- getIntParmCPLEX, 127, 132
- getIntQualCPLEX, 128
- getItCntCPLEX, 129
- getLogFileCPLEX, 130, 216
- getLogFileNameCPLEX, 131
- getLongParmCPLEX, 132
- getLowBndsIdsCPLEX, 133
- getLowerBndsCPLEX, 28, 133, 134
- getMethodCPLEX, 135
- getMIPrelGapCPLEX, 136
- getMIPstartIndexCPLEX, 137
- getMIPstartNameCPLEX, 138
- getMIPstartsCPLEX, 139
- getNumColsCPLEX, 29, 140
- getNumMIPstartsCPLEX, 141
- getNumNnzCPLEX, 142
- getNumQConstrsCPLEX, 143
- getNumQPnzCPLEX, 144
- getNumQuadCPLEX, 145

- getNumRowsCPLEX, 29, 146
- getObjCPLEX, 147
- getObjDirCPLEX, 148
- getObjNameCPLEX, 149
- getObjOffsetCPLEX, 150
- getObjValCPLEX, 136, 151
- getOrderCPLEX, 152
- getParmHierNameCPLEX, 153
- getParmNameCPLEX, 154
- getParmNumCPLEX, 155
- getParmTypeCPLEX, 156
- getParmValCPLEX, 82, 157
- getPhase1CntCPLEX, 157
- getPicCPLEX, 158
- getPreStatCPLEX, 159
- getProbNameCPLEX, 160
- getProbTypeCPLEX, 38, 161, 225
- getProbVarCPLEX, 162
- getQConstrCPLEX, 163
- getQPcoefCPLEX, 164
- getQuadCPLEX, 165
- getRedLpCPLEX, 166
- getRhsCPLEX, 167
- getRngValCPLEX, 168
- getRowIndexCPLEX, 169
- getRowInfeasCPLEX, 98, 170
- getRowNameCPLEX, 171
- getRowsCPLEX, 172
- getSenseCPLEX, 173
- getSiftItCntCPLEX, 174
- getSiftPase1CntCPLEX, 175
- getSlackCPLEX, 176
- getStatCPLEX, 15, 96, 98, 177, 186, 187, 189, 190, 199, 201, 226
- getStatStrCPLEX, 120, 178, 226
- getStrParmCPLEX, 179
- getSubMethodCPLEX, 180
- getSubStatCPLEX, 181
- getTimeCPLEX, 182
- getUppBndsIdsCPLEX, 183
- getUpperBndsCPLEX, 28, 183, 184
- getVersionCPLEX, 185

- hybbaroptCPLEX, 185
- hybnetoptCPLEX, 186

- initProbCPLEX, 8, 10–23, 25–43, 45, 46, 49–59, 84, 86–93, 96–98, 102–104, 107–115, 117–119, 121, 122, 128,

- [129, 133–152, 158–177, 180, 181, 183, 184, 186, 187, 187, 188–192, 195–199, 201–203, 205–210, 212, 219, 222–225, 227–230, 232](#)
 isCPLEXchanPointer (cplexPtr-class), [83](#)
 isCPLEXchanPointer, cplexPtr-method (cplexPtr-class), [83](#)
 isCPLEXenvPointer (cplexPtr-class), [83](#)
 isCPLEXenvPointer, cplexPtr-method (cplexPtr-class), [83](#)
 isCPLEXfilePointer (cplexPtr-class), [83](#)
 isCPLEXfilePointer, cplexPtr-method (cplexPtr-class), [83](#)
 isCPLEXprobPointer (cplexPtr-class), [83](#)
 isCPLEXprobPointer, cplexPtr-method (cplexPtr-class), [83](#)
 isCPLEXtermPointer (cplexPtr-class), [83](#)
 isCPLEXtermPointer, cplexPtr-method (cplexPtr-class), [83](#)
 isNULLpointerCPLEX (cplexPtr-class), [83](#)
 isNULLpointerCPLEX, cplexPtr-method (cplexPtr-class), [83](#)
- lpoptCPLEX, [188](#)
 mipoptCPLEX, [189](#)
- newColsCPLEX, [190](#)
 newRowsCPLEX, [191](#)
- objSaCPLEX, [192](#)
 openEnvCPLEX, [8–23, 25–47, 49–59, 84–98, 100–192, 193, 195–210, 212–232](#)
 openFileCPLEX, [9, 47, 85, 99, 182, 194, 216](#)
 openProbCPLEX, [48, 195](#)
 ordWriteCPLEX, [196](#)
- preslvWriteCPLEX, [197](#)
 presolveCPLEX, [198](#)
 primoptCPLEX, [199](#)
 printTerminateCPLEX, [44, 94, 200, 221](#)
- qpoptCPLEX, [200](#)
- readCopyBaseCPLEX, [201](#)
 readCopyMIPstartsCPLEX, [202](#)
 readCopyOrderCPLEX, [203](#)
 readCopyParmCPLEX, [204](#)
 readCopyProbCPLEX, [197, 205](#)
 readCopySolCPLEX, [206](#)
- refineConflictCPLEX, [207](#)
 refineConflictExtCPLEX, [208](#)
 refineMIPstartConflictCPLEX, [209](#)
 refineMIPstartConflictExtCPLEX, [210](#)
 return_codeCPLEX, [82, 211](#)
 rhsSaCPLEX, [212](#)
- setDblParmCPLEX, [213](#)
 setDefaultParmCPLEX, [214](#)
 setIntParmCPLEX, [215, 218](#)
 setLogFileCPLEX, [130, 216](#)
 setLogFileNameCPLEX, [217](#)
 setLongParmCPLEX, [218](#)
 setObjDirCPLEX, [219](#)
 setStrParmCPLEX, [220](#)
 setTerminateCPLEX, [44, 94, 200, 221](#)
 siftoptCPLEX, [222](#)
 solnInfoCPLEX, [15, 83, 96, 98, 186, 187, 189, 190, 199, 201, 223, 224](#)
 solutionCPLEX, [15, 83, 96, 98, 186, 187, 189, 190, 199, 201, 223, 224](#)
 solWriteCPLEX, [225](#)
 status_codeCPLEX, [82, 226](#)
 summary, cplexPtr-method (cplexPtr-class), [83](#)
- tightenBndsCPLEX, [32, 227](#)
 tuneParmCPLEX, [228](#)
- unscaleProbCPLEX, [229](#)
- writeMIPstartsCPLEX, [230](#)
 writeParmCPLEX, [231](#)
 writeProbCPLEX, [232](#)