NAME
CUTEST_ccf – CUTEst tool to evaluate constraint functions values.

SYNOPSIS

CALL CUTEST_ccf(status, n, m, X, C)

For real rather than double precision arguments, instead

CALL CUTEST_ccf_s(...)

and for quadruple precision arguments, when available,

CALL CUTEST_ccf_q(...)

DESCRIPTION
The CUTEST_ccf subroutine evaluates the values of the constraint functions of the problem decoded from a SIF file by the script *sifdecoder* at the point X.

The problem under consideration is to minimize or maximize an objective function $f(x)$ over all $x \in R^n$ subject to general equations $c_i(x) = 0$, $(i \in 1, \ldots, m_E)$, general inequalities $c_i^l \le c_i(x) \le c_i^{u,}$ $(i \in m_E + 1, \ldots, m)$, and simple bounds $x^l \le x \le x^{u.}$ The objective function is group-partially separable and all constraint functions are partially separable.

ARGUMENTS
The arguments of CUTEST_ccf are as follows

status [out] - integer
the outputr status: 0 for a succesful call, 1 for an array allocation/deallocation error, 2 for an array bound error, 3 for an evaluation error,

n [in] - integer
the number of variables for the problem,

m [in] - integer
the total number of general constraints,

X [in] - real/double precision
an array which gives the current estimate of the solution of the problem,

C [out] - real/double precision
an array which gives the values of the general constraint functions evaluated at X. The i-th component of C will contain the value of $c_i(x)$.

AUTHORS
J.M. Fowkes, N.I.M. Gould, A. Montoison and D. Orban

SEE ALSO
CUTEst: a Constrained and Unconstrained Testing Environment with safe threads,
N.I.M. Gould, D. Orban and Ph.L. Toint,
Computational Optimization and Applications **60**:3, pp.545-557, 2014.

CUTEr (and SifDec): A Constrained and Unconstrained Testing Environment, revisited,
  N.I.M. Gould, D. Orban and Ph.L. Toint,
  ACM TOMS, **29**:4, pp.373-394, 2003.

CUTE: Constrained and Unconstrained Testing Environment,
  I. Bongartz, A.R. Conn, N.I.M. Gould and Ph.L. Toint,
  ACM TOMS, **21**:1, pp.123-160, 1995.

sifdecoder(1)