

Help

```

#if defined(PremiaCurrentVersion) && PremiaCurrentVersion <
    (2008+2) //The "#else" part of the code will be freely av
    ailable after the (year of creation of this file + 2)
#else
/*****
    *****/
/*
    */
/*****
    *****/
/*
    */
/* Multi-Level SOLVers
    */
/*
    */
/* Copyright (C) 1992-1995 Tomas Skalicky. All rights res
    erved.
    */
/*
    */
/*****
    *****/
/*
    */
/*      ANY USE OF THIS CODE CONSTITUTES ACCEPTANCE OF TH
    E TERMS
    */
/*      OF THE COPYRIGHT NOTICE (SEE FILE copyright.h
    )
    */
/*
    */
/*****
    *****/

#endif
#define MLSOLV_H

#include "highdim_vector.h"
#include "highdim_matrix.h"
#include "qmatrix.h"
#include "itersolv.h"

```

```

#include "copyright.h"

Vector *MGStep(int NoLevels, QMatrix *A, Vector *x, Vector
    *b,
                Matrix *R, Matrix *P, int Level, int Gamma,
                IterProcType SmoothProc, int Nu1, int Nu2,
                PrecondProcType PrecondProc, double Omega,
                IterProcType SolvProc, int NuC,
                PrecondProcType PrecondProcC, double OmegaC)
    ;
Vector *MGIter(int NoLevels, QMatrix *A, Vector *x, Vector
    *b,
                Matrix *R, Matrix *P, int MaxIter, int Gam
    ma,
                IterProcType SmoothProc, int Nu1, int Nu2,
                PrecondProcType PrecondProc, double Omega,
                IterProcType SolvProc, int NuC,
                PrecondProcType PrecondProcC, double OmegaC)
    ;
Vector *NestedMGIter(int NoLevels, QMatrix *A, Vector *x,
    Vector *b,
                Matrix *R, Matrix *P, int Gamma,
                IterProcType SmoothProc, int Nu1, int
    Nu2,
                PrecondProcType PrecondProc, double Om
    ega,
                IterProcType SolvProc, int NuC,
                PrecondProcType PrecondProcC, double
    OmegaC);
Vector *MGPCGIter(int NoLevels, QMatrix *A, Vector *x, Vec
    tor *b,
                Matrix *R, Matrix *P, int MaxIter, int
    NoMGIter, int Gamma,
                IterProcType SmoothProc, int Nu1, int Nu2
    ,
                PrecondProcType PrecondProc, double Omeg
    a,
                IterProcType SolvProc, int NuC,
                PrecondProcType PrecondProcC, double Omeg
    aC);
Vector *BPXPrecond(int NoLevels, QMatrix *A, Vector *y, Vec

```

```

    tor *c,
        Matrix *R, Matrix *P, int Level,
        IterProcType SmoothProc, int Nu,
        PrecondProcType PrecondProc, double Omeg
    a,
        IterProcType SmoothProcC, int NuC,
        PrecondProcType PrecondProcC, double Om
    egaC);
Vector *BPXPCGIter(int NoLevels, QMatrix *A, Vector *x, Vec
    tor *b,
        Matrix *R, Matrix *P, int MaxIter,
        IterProcType SmoothProc, int Nu,
        PrecondProcType PrecondProc, double Omeg
    a,
        IterProcType SmoothProcC, int NuC,
        PrecondProcType PrecondProcC, double Om
    egaC);

#endif /* MLSOLV_H */

#endif //PremiaCurrentVersion

```

References