```
    Help
#ifndef _WIENERHOPFRS_H
#define _WIENERHOPFRS_H

#include <stdlib.h>
#include <math.h>

#include "pnl/pnl_vector.h"
#include "pnl/pnl_matrix.h"
#include "pnl/pnl_fft.h"
//#include  "vector.h"
//#include  "dft.h"




/*//////////////////////////////////////////////////*/
void realfastfouriertransform(PnlVect *a, int tnn, int inv
    erseffft);

int readparamskou_rs(int *nstates, PnlVect **rr, PnlVect **
    divi, PnlVect **sigmas, PnlVect **lambdam, PnlVect **lambd
    ap, PnlVect **lambda, PnlVect **pp,  PnlMat **lam, char *fil
    ename);

int readparamstsl_rs(int *nstates, PnlVect **rr, PnlVect **
    divi, PnlVect **nums, PnlVect **nups, PnlVect **lambdam, Pn
    lVect **lambdap, PnlVect **cm, PnlVect **cp, PnlMat **lam,
    char *filename);

int fastwienerhopfamerican_rs(int model, long int Nr, PnlV
    ect *mu, PnlVect *qu, double om,
    int ifCall, double Spot, PnlVect *lm1, PnlVect *lp1,
    PnlVect *num,PnlVect *nup, PnlVect *cnum,PnlVect *cnup,
    PnlVect *r, PnlVect *divid, PnlMat *lam,
    double T, double h, PnlVect *Strike1,
    double er, long int step, double eps,
    PnlVect *ptprice, PnlVect *ptdelta);

int fastwienerhopf_rs(int model, long int Nr, PnlVect *mu,
    PnlVect *qu, double om, int am, int upordown,
    int ifCall, double Spot, PnlVect *lm1, PnlVect *lp1,
```

```
    PnlVect *num,PnlVect *nup, PnlVect *cnum,PnlVect *cnup,
    PnlVect *r, PnlVect *divid, PnlMat *lam,
    double T, double h, PnlVect *Strike1,
  double bar, PnlVect *rebate,
    double er, long int step, double eps,
    PnlVect *ptprice, PnlVect *ptdelta);
int fastwienerhopf_hs(int model, long int Nr, PnlVect *mu,
    PnlVect *qu, double om, int am, int upordown,
    int ifCall, double Spot, double lm1, double lp1,
    PnlVect *sg, PnlVect *num, PnlVect *nup, double cnum,
    double cnup,
    double r, double divid, PnlMat *lam,
    double T, double h, double Strike1,
  double bar, double rebate,
    double er, long int step, PnlVect *ptprice, PnlVect *pt
    delta);
int fastwienerhopfamer_hs(int model, long int Nr, PnlVect *
    mu, PnlVect *qu, double om,
    int ifCall, double Spot, double lm1, double lp1,
    PnlVect *sg, PnlVect *num, PnlVect *nup, double cnum,
    double cnup,
    double r, double divi, PnlMat *lam,
    double T, double h, double Strike1,
  double er, long int step, PnlVect *ptprice, PnlVect *pt
    delta);

#endif
```

References