Help

```
#ifndef _ALFONSI_H
#define _ALFONSI_H

#include "optype.h"
#include "pnl/pnl_mathtools.h"
#include "pnl/pnl_random.h"
#include "pnl/pnl_cdf.h"

/*////////////////////////////////////////////////*/
double psik (double t, double k);
double DiscLawMatch5(int generator);
double DiscLawMatch7(int generator);
void Heston01(double *x1, double *x2, double *x3, double *x
    4, double dt, double dw,double a, double k, double sig,
    double mu, double rho, double Kseuil,int generator,int flag_cir)
    ;
void Heston02 (double *x1, double *x3,double dw2, double rh
    o);
void fct_Heston(double *x1, double *x2, double *x3, double
    *x4, double dt, double dw,double dw2,double a, double k,
    double sig, double mu, double rho, double Kseuil,int generator,i
    nt flag_cir);

/* see alfonsi.c*/
int HestonSimulation_Alfonsi(int flag_SpotPaths, PnlMat *
    SpotPaths, int flag_VarPaths, PnlMat *VarPaths, int flag_Av
    eragePaths, PnlMat *AveragePaths, double S0, double T,
    double r, double divid, double V0,double k,double theta,double
    sigma,double rho, long NbrMCsimulation, int NbrDates, int
    NbrStepPerPeriod, int generator,int flag_cir);

int HestonSimulation_Alfonsi_Modified(int flag_SpotPaths,
    PnlMat *SpotPaths, int flag_VarPaths, PnlMat *VarPaths,
    int flag_AveragePaths, PnlMat *AveragePaths,PnlMat *VarianceI
    nt, double S0, double T, double r, double divid,double V0,
    double k, double theta,double sigma,double rho,long NbrMCsimulat
    ion, int NbrDates, int NbrStepPerPeriod,int generator,int
    flag_cir);

/* see alfonsi.c*/
```

```
int BatesSimulation_Alfonsi (int flag_SpotPaths, PnlMat *
    SpotPaths, int flag_VarPaths, PnlMat *VarPaths, int flag_Av
    eragePaths, PnlMat *AveragePaths, double S0, double T,
    double r, double divid, double V0, double k, double theta,
    double sigma, double rho, double mu_jump, double gamma2, double
    lambda, long NbrMCsimulation, int NbrDates, int NbrStepPerP
    eriod, int generator, int flag_cir);

/* Functions used in the regression basis in Longstaff-Sch
    wartz algorithm*/
// Approximation formula for a european option under
    Heston model.
int ApAntonelliScarlattiHeston(double S, NumFunc_1  *p,
    double T, double r, double divid, double v0,double kappa,double
    theta,double sigma,double rho,double *ptprice, double *ptde
    lta);


// Approximation formula for a european option under
    Heston model.
int ApAlosHeston(double S, NumFunc_1  *p, double T, double
    r, double divid, double v0,double kappa,double theta,
    double sigma,double rho,double *ptprice, double *ptdelta);

// Approximation formula for a european option under Bates model.
int ApAlosBates(double S, NumFunc_1  *p, double T, double
    r, double divid, double v0,double kappa,double theta,
    double sigma,double rho,double m,double v,double lambda,double *
    ptprice, double *ptdelta);

// Approximation formula for a european asian-option under     Black-Scholes mod
int Ap_FixedAsian_BlackScholes(double Current_Spot, double
    Current_Avg, double Current_Date, NumFunc_2  *p, double Maturity,
     double r, double divid, double sigma, double *ptprice,
    double *ptdelta);

#endif
```

References