```
   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

double Asian_BS_FusaiMeucci(double spot, double strike,
           double maturity, double rf, double dividend
    ,
           double sigmaBS,
           int nmonitoringdates,
           double lowlim, double uplim,
                     int nquadpoints, long nfft,
                            double price[], double solutio
    n[],double *delta);

double Asian_NIG_FusaiMeucci(double spot,
           double strike,
           double maturity,
                    double rf,
           double dividend,
           double alphaNIG, double betaNIG,double delt
    aNIG,
           int nmonitoringdates,
           double lowlim,
           double uplim,
                     int nquadpoints,        //n. of qu
    adrature points
           long nfft,
           double price[],
                     double solution[],double *delta);

double Asian_MERTON_FusaiMeucci(double spot, double strike,

           double maturity, double rf, double dividend
    ,
           double sgMerton, double alphaMerton,
    double lambdaMerton, double deltaMerton,
           int nmonitoringdates,
           double lowlim,  double uplim,
                     int nquadpoints, long nfft,
```

```
                    double price[], double solution[],
    double *delta);

double Asian_CGMY_FusaiMeucci(double spot,
           double strike,
           double maturity,
                   double rf,
           double dividend,
           double CCGMY, double GCGMY, double MCGMY,
    double YCGMY,
           int nmonitoringdates,
           double lowlim,
           double uplim,
                     int nquadpoints,        //n. of qu
    adrature points
           long nfft,
           double price[],
           double solution[],double *delta);

double Asian_DE_FusaiMeucci(double spot,
           double strike,
           double maturity,
                   double rf,
           double dividend,
           double sgDE, double lambdaDE, double pDE,
    double eta1DE, double eta2DE,
           int nmonitoringdates,
           double lowlim,
           double uplim,
                     int nquadpoints,        //n. of qu
    adrature points
           long nfft,
           double price[],
                           double solution[],double *delt
    a);
//OUTPUT: Contains the solution
double DiscreteAsian(int model,          //modello
                     double spot,
           double strike,
                   double rf,
           double dt,
```

```
            int ndates,
            double lowlim,
            double uplim,
                      int npoints,        //n. of quadra
    ture points
            long nfft,          //n. of points fo
    r the fft inversion
            double ModelParameters[],  //the paramete
    rs of the model
            double price[],
            double solution[],double *delta);      /
    /OUTPUT: Contains the solution


//compute the moments of L
void newmomentsAM(int model, double rf, double dt, int max
    moment,
          int ndates, double parameters[], double **
    momtable);

//compute the moments of the arithemtic average given the
    moments of L
void newmomentsArithM(int ndates, double Lmoments[],
    double *AvgMoments);

//compute the probability bound
//using the moment bound
double boundAM(int model, double bound, double rf, double
    dt, int maxmoment,
          int ndates, double parameters[], double
    moments[]);

//We find in an  authomatic way the extremes of integration
int findlowuplimit(int model,  double rf, double dt, int
    maxnummoments,
          int ndates, double lowfactor, double up
    factor,
          double parameters[], double extremes[]);

#endif //PremiaCurrentVersion
```

References