```
   Help
extern "C"{
#include "temperedstable1d_vol.h"
}
#include "math/numerics.h"
extern "C"{

#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)
static int CHK_OPT(CF_CGMY_VARIANCESWAP)(void *Opt, void *
    Mod)
{
  return NONACTIVE;
}
int CALC(CF_CGMY_VARIANCESWAP)(void *Opt,void *Mod,Pricing
    Method *Met)
{
return AVAILABLE_IN_FULL_PREMIA;
}
#else
  //-----------------------------------------------------
    -------------
  static int ap_cgmy_varswap_cf(double S0, double Strike,
    double T, double r, double divid, double ap, double am,double
    lap,double lam,double cpp,double cmm, double *fairval,
    double *ptprice)
  {
    double K;

   double gamma2p, gamma2m;

    K=Strike;

    gamma2p=tgamma(2.0-ap);
    gamma2m=tgamma(2.0-am);
    double lpnu=exp((2.0-ap)*log(lap));
    double lmnu=exp((2.0-am)*log(lam));

    double mval=cpp*gamma2p/lpnu+cmm*gamma2m/lmnu;
```

```
    *fairval = sqrt(mval)*100.0;
    *ptprice= exp(-r*T)*(mval*10000-K*K);

    return OK;
  }


  int CALC(CF_CGMY_VARIANCESWAP)(void *Opt,void *Mod,Prici
    ngMethod *Met)
  {
    TYPEOPT* ptOpt=(TYPEOPT*)Opt;
    TYPEMOD* ptMod=(TYPEMOD*)Mod;
    double r, divid, strike, spot;
    NumFunc_1 *p;

    r=log(1.+ptMod->R.Val.V_DOUBLE/100.);
    divid=log(1.+ptMod->Divid.Val.V_DOUBLE/100.);
    p=ptOpt->PayOff.Val.V_NUMFUNC_1;
    strike=p->Par[0].Val.V_DOUBLE;
    spot=ptMod->S0.Val.V_DOUBLE;

    return ap_cgmy_varswap_cf(
      spot, strike, ptOpt->Maturity.Val.V_DATE-ptMod->T.Val
    .V_DATE, r, divid,  ptMod->AlphaPlus.Val.V_PDOUBLE, ptMod-
    >AlphaMinus.Val.V_PDOUBLE, ptMod->LambdaPlus.Val.V_PDOUB
    LE, ptMod->LambdaMinus.Val.V_PDOUBLE, ptMod->CPlus.Val.V_PDO
    UBLE, ptMod->CMinus.Val.V_PDOUBLE,
    &(Met->Res[0].Val.V_DOUBLE), &(Met->Res[1].Val.V_
    DOUBLE));
  }

  static int CHK_OPT(CF_CGMY_VARIANCESWAP)(void *Opt, void
    *Mod)
  {
    if ((strcmp( ((Option*)Opt)->Name,"VarianceSwap")==0))
      return OK;

    return WRONG;
  }

#endif //PremiaCurrentVersion
```

```
  static int MET(Init)(PricingMethod *Met,Option *Opt)
  {
    static int first=1;

    if (first)
    {
      first=0;
    }
    return OK;
  }

  PricingMethod MET(CF_CGMY_VARIANCESWAP)=
  {
    "CF_CGMY_VARIANCESWAP",
    { {" ",PREMIA_NULLTYPE,{0},FORBID}},
    CALC(CF_CGMY_VARIANCESWAP),
    {   {"Fair strike in annual volatility points",DOUBLE,{
    100},FORBID},
        {"Price in 10000 variance points",DOUBLE,{100},FORB
    ID},
        {" ",PREMIA_NULLTYPE,{0},FORBID}},
    CHK_OPT(CF_CGMY_VARIANCESWAP),
    CHK_ok ,
    MET(Init)
  } ;

  /*////////////////////////////////////*/
}
```

References