Help

```
#include  "
href../../../mod/sg1d/sg1d_stdi/sg1d_stdi_h_src.pdfsg1d_stdi.h"
#include  "
href../../../mod/sg1d/sg1d_stdi/QuadraticModel_h_src.pdfQuadraticModel.h"
#include "
href../../../common/math/read_market_zc/InitialYieldCurve_h_src.pdfmath/read_mar


//The "#else" part of the code will be freely available after the (year of creat
#if defined(PremiaCurrentVersion) && PremiaCurrentVersion < (2007+2)
int CALC(CF_ZCBondSG1D)(void *Opt, void *Mod, PricingMethod *Met)
{
  return AVAILABLE_IN_FULL_PREMIA;
}
static int CHK_OPT(CF_ZCBondSG1D)(void *Opt, void *Mod)
{
  return NONACTIVE;
}
#else

/*Call Option*/
static int zcb_quad1d(double flat_flag, double beta, double sigma, double r0, ch
{
  double x0;
  Data data;

  ZCMarketData ZCMarket;

  /* Flag to decide to read or not ZC bond datas in "initialyields.dat" */
  /* If P(0,T) not read then P(0,T)=exp(-r0*T) */
  if (flat_flag == 0)
    {
      ZCMarket.FlatOrMarket = 0;
      ZCMarket.Rate = r0;
    }

  else
    {
      ZCMarket.FlatOrMarket = 1;
```

```
      ZCMarket.filename = curve;
      ReadMarketData(&ZCMarket);

      r0 = -log(BondPrice(INC, &ZCMarket)) / INC;

      if (T > GET(ZCMarket.tm, ZCMarket.Nvalue - 1))
        {
          printf("\ nError : time bigger than the last time value entered in ini
          exit(EXIT_FAILURE);
        }
    }

  x0 = sqrt(2.*r0);
  /* coefficients of P(0,T) */
  bond_coeffs(&ZCMarket, &data, T, beta, sigma, x0);

  /*Price*/
  *price = exp(-(r0 * data.B + data.b * x0 + data.c));

  DeleteZCMarketData(&ZCMarket);

  return OK;
}

int CALC(CF_ZCBondSG1D)(void *Opt, void *Mod, PricingMethod *Met)
{
  TYPEOPT *ptOpt = (TYPEOPT *)Opt;
  TYPEMOD *ptMod = (TYPEMOD *)Mod;

  return zcb_quad1d(ptMod->flat_flag.Val.V_INT,
                    ptMod->a.Val.V_DOUBLE,
                    ptMod->Sigma.Val.V_PDOUBLE,
                    MOD(GetYield)(ptMod),
                    MOD(GetCurve)(ptMod),
                    ptOpt->BMaturity.Val.V_DATE - ptMod->T.Val.V_DATE,
                    &(Met->Res[0].Val.V_DOUBLE));
}

static int CHK_OPT(CF_ZCBondSG1D)(void *Opt, void *Mod)
{
  return strcmp(((Option *)Opt)->Name, "ZeroCouponBond");
```

```
}

#endif //PremiaCurrentVersion
static int MET(Init)(PricingMethod *Met, Option *Opt)
{
  if (Met->init == 0)
    {
      Met->init = 1;
      Met->HelpFilenameHint = "cf_quadratic1d_zcbond";
    }

  return OK;
}

PricingMethod MET(CF_ZCBondSG1D) =
{
  "CF_SquareGaussian1d_ZCBond",
  {{" ", PREMIA_NULLTYPE, {0}, FORBID}},
  CALC(CF_ZCBondSG1D),
  {{"Price", DOUBLE, {100}, FORBID} , {" ", PREMIA_NULLTYPE, {0}, FORBID}},
  CHK_OPT(CF_ZCBondSG1D),
  CHK_ok,
  MET(Init)
} ;
```