```
   Help
#include "bs1d_doublim.h"
#define INC 1.0e-5 /*Relative Increment for Delta-Hedging*/

static int CallIn_KunitomoIkeda_91(double s,NumFunc_1 *L,
    NumFunc_1 *U,NumFunc_1  *Rebate,NumFunc_1 *PayOff,double t,
    double r,double divid,double sigma,double *ptprice,double *ptde
    lta){

  double price,delta,out_price,out_delta,price_plus,price_m
    inus;

  pnl_cf_call_bs(s,PayOff->Par[0].Val.V_PDOUBLE,t,r,divid,
    sigma,&price,&delta);
  CallOut_KunitomoIkeda_91(s,L,U,Rebate,PayOff,t,r,divid,si
    gma,&out_price,&out_delta);

  /*Price*/
  *ptprice=price-out_price;

  pnl_cf_call_bs(s*(1.+INC),PayOff->Par[0].Val.V_PDOUBLE,t,
    r,divid,sigma,&price,&delta);
  CallOut_KunitomoIkeda_91(s*(1.+INC),L,U,Rebate,PayOff,t,
    r,divid,sigma,&out_price,&out_delta);
  price_plus=price-out_price;

  pnl_cf_call_bs(s*(1.-INC),PayOff->Par[0].Val.V_PDOUBLE,t,
    r,divid,sigma,&price,&delta);
  CallOut_KunitomoIkeda_91(s*(1.-INC),L,U,Rebate,PayOff,t,
    r,divid,sigma,&out_price,&out_delta);
  price_minus=price-out_price;

  /*Delta*/
  *ptdelta=(price_plus-price_minus)/(2.*s*INC);

  return OK;
}

int CALC(CF_CallIn_KunitomoIkeda)(void*Opt,void *Mod,Prici
    ngMethod *Met)
{
  TYPEOPT* ptOpt=(TYPEOPT*)Opt;
  TYPEMOD* ptMod=(TYPEMOD*)Mod;
  double r,divid;

  r=log(1.+ptMod->R.Val.V_DOUBLE/100.);
  divid=log(1.+ptMod->Divid.Val.V_DOUBLE/100.);

  return  CallIn_KunitomoIkeda_91(ptMod->S0.Val.V_PDOUBLE,
    ptOpt->LowerLimit.Val.V_NUMFUNC_1, ptOpt->UpperLimit.Val.V_
    NUMFUNC_1, ptOpt->Rebate.Val.V_NUMFUNC_1,ptOpt->PayOff.Val.V_
    NUMFUNC_1,ptOpt->Maturity.Val.V_DATE-ptMod->T.Val.V_DATE,r,divid,
    ptMod->Sigma.Val.V_PDOUBLE,&(Met->Res[0].Val.V_DOUBLE),&(
    Met->Res[1].Val.V_DOUBLE));
}

static int CHK_OPT(CF_CallIn_KunitomoIkeda)(void *Opt, voi
    d *Mod)
{
  Option* ptOpt=(Option*)Opt;
  TYPEOPT* opt=(TYPEOPT*)(ptOpt->TypeOpt);


  if ((opt->Parisian).Val.V_BOOL==WRONG)
    if((opt->RebOrNo).Val.V_BOOL==NOREBATE)
      return strcmp(((Option*)Opt)->Name,"    DoubleCallInEuro");
  return WRONG;

}

static int MET(Init)(PricingMethod *Met,Option *Opt)
{
  return OK;
}

PricingMethod MET(CF_CallIn_KunitomoIkeda)=
{
  "CF_CallIn_KunitomoIkeda",
  {{" ",PREMIA_NULLTYPE,{0},FORBID}},
  CALC(CF_CallIn_KunitomoIkeda),
  {{"Price",DOUBLE,{100},FORBID},{"Delta",DOUBLE,{100},FORB
    ID} ,{" ",PREMIA_NULLTYPE,{0},FORBID}},
  CHK_OPT(CF_CallIn_KunitomoIkeda),
  CHK_ok,
  MET(Init)
} ;
```

References