

[Help](#)

```
#include "doublim.h"

static NumFunc_1 call=
{
    Call,
    {"Strike",PDOUBLE,100,ALLOW},{"  ",END,0,FORBID}},
    CHK_call
};

static NumFunc_1 const_Re=
{
    Const,
    {"Const Rebate",DOUBLE,100,ALLOW}, {"  ",
    END,0,FORBID}},
    CHK_ok
};

static NumFunc_1 const_Low=
{
    Const,
    {"Lower Limit",PDOUBLE,100,ALLOW},
    {"Delay",SPDOUBLE,0,ALLOW},
    {"  ",END,0,FORBID}},
    CHK_call
};

static NumFunc_1 const_Up=
{
    Const,
    {"Upper Limit",PDOUBLE,100,ALLOW},
    {"Delay",SPDOUBLE,0,ALLOW},
    {"  ",END,0,FORBID}},
    CHK_call
};

static TYPEOPT ParisianDoubleCallInEuro=
{
    /*PayOff*/          {"PayOff",NUMFUNC_1,0,FO
```

```

    RBID},
    /*Rebate*/          {"Const Rebate", NUMFUNC_1
    ,0,FORBID},
    /*LowerLimit*/      {"Lower Limit", NUMFUNC_1,
    0,FORBID},
    /*UpperLimit*/      {"Upper Limit", NUMFUNC_1,
    0,FORBID},
    /*OutOrIn*/          {"In", BOOL, IN, FORBID},
    /*Parisian*/         {"Parisian", BOOL, 0, FORBID},
    /*RebNo*/            {"Rebate", BOOL, REBATE, FORB
    ID},
    /*EuOrAm*/           {"Euro", BOOL, EURO, FORBID},
    /*Maturity*/         {"Maturity", DATE, 0, ALLOW}
    };

static int OPT(Init)(Option *opt)
{
    TYPEOPT* pt=( TYPEOPT*)(opt->TypeOpt);
    static int first=1;

    if (first)
    {
        pt->PayOff.Val.V_NUMFUNC_1=&call;
        pt->Rebate.Val.V_NUMFUNC_1=&const_Re;
        pt->LowerLimit.Val.V_NUMFUNC_1=&const_Low
    ;
        pt->UpperLimit.Val.V_NUMFUNC_1=&const_Up;

        (pt->EuOrAm).Val.V_BOOL=EURO;
        (pt->OutOrIn).Val.V_BOOL=IN;
        (pt->RebOrNo).Val.V_BOOL=NOREBATE;
        (pt->Maturity).Val.V_DATE=1.0;

        (pt->PayOff.Val.V_NUMFUNC_1)->Par[0].Val.
        V_PDOUBLE=100.0;
        (pt->Rebate.Val.V_NUMFUNC_1)->Par[0].Val.
        V_PDOUBLE=0.0;
        (pt->LowerLimit.Val.V_NUMFUNC_1)->Par[0].
        Val.V_PDOUBLE=90.0;
        (pt->UpperLimit.Val.V_NUMFUNC_1)->Par[0].
        Val.V_PDOUBLE=110.0;
    }
}

```

```
        (pt->LowerLimit.Val.V_NUMFUNC_1)->Par[1].
Val.V_SPDOUBLE=0.01;
        (pt->UpperLimit.Val.V_NUMFUNC_1)->Par[1].
Val.V_SPDOUBLE=0.01;
        first=0;
    }

    return OK;
}

MAKEOPT(ParisianDoubleCallInEuro);
```

References