

## [Help](#)

```
#include "  
href../../mod/libor_affine_gould/libor_affine_gould_h_src.pdflibor_affine_gould.  
#include "  
href../../common/chk_h_src.pdfchk.h"  
#include "  
href../../mod/hes1d/hes1d_pad/model_h_src.pdfmodel.h"  
#include "  
href../../common/enums_h_src.pdfenums.h"  
#include "premia_obj.h"
```

```
double MOD(GetYield)(TYPEMOD *pt)  
{  
    VAR *Par;  
    Par = lookup_premia_enum_par(&(pt->flat_flag), 0);  
    return Par[0].Val.V_PDOUBLE;  
}
```

```
char *MOD(GetCurve)(TYPEMOD *pt)  
{  
    VAR *Par;  
    Par = lookup_premia_enum_par(&(pt->flat_flag), 1);  
    return Par[0].Val.V_FILENAME;  
}
```

```
static int MOD(Init)(Model *model)  
{  
    VAR *Par;  
    TYPEMOD *pt = (TYPEMOD *)(model->TypeModel);  
  
    if (model->init == 0)  
    {  
        model->init = 1;  
        model->nvar = 0;  
  
        pt->T.Vname = "Current Date";  
        pt->T.Vtype = DATE;  
        pt->T.Val.V_DATE = 0.0;  
        pt->T.Viter = ALLOW;
```

```

model->nvar++;

pt->flat_flag.Vname = "Initial Yield Curve";
pt->flat_flag.Vtype = ENUM;
pt->flat_flag.Val.V_ENUM.value = 0;
pt->flat_flag.Val.V_ENUM.members = &PremiaEnumFlat;
pt->flat_flag.Viter = ALLOW;
model->nvar++;
Par = lookup_premia_enum_par(&(pt->flat_flag), 0);
Par[0].Vname = "Yield Value";
Par[0].Vtype = PDOUBLE;
Par[0].Val.V_PDOUBLE = 0.03;
Par[0].Viter = ALLOW;
Par = lookup_premia_enum_par(&(pt->flat_flag), 1);
Par[0].Vname = "Yield Curve";
Par[0].Vtype = FILENAME;
Par[0].Val.V_FILENAME = NULL;
Par[0].Viter = FORBID;

pt->x0.Vname = "x0";
pt->x0.Vtype = PDOUBLE;
pt->x0.Val.V_PDOUBLE = 1.25;
pt->x0.Viter = ALLOW;
model->nvar++;

pt->lambda.Vname = "lambda";
pt->lambda.Vtype = PDOUBLE;
pt->lambda.Val.V_PDOUBLE = 0.01;
pt->lambda.Viter = ALLOW;
model->nvar++;

pt->alpha.Vname = "alpha";
pt->alpha.Vtype = PDOUBLE;
pt->alpha.Val.V_PDOUBLE = 2.0;
pt->alpha.Viter = ALLOW;
model->nvar++;

pt->beta.Vname = "beta";
pt->beta.Vtype = PDOUBLE;
pt->beta.Val.V_PDOUBLE = 1.0;
pt->beta.Viter = ALLOW;

```

```

        model->nvar++;
    }

    Par = lookup_premia_enum_par(&(pt->flat_flag), 1);
    if (Par[0].Val.V_FILENAME == NULL)
    {
        if ((Par[0].Val.V_FILENAME = malloc(sizeof(char) * MAX_PATH_LEN)) == NULL)
            return MEMORY_ALLOCATION_FAILURE;
        sprintf(Par[0].Val.V_FILENAME, "%s%sinitialyield.dat", premia_data_dir, pa
    }
    return OK;
}

```

```

TYPEMOD Libor_Affine_Gould;
MAKEMOD(Libor_Affine_Gould);

```