```
   Help
#if defined(PremiaCurrentVersion) && PremiaCurrentVersion <
     (2011+2) //The "#else" part of the code will be freely av
    ailable after the (year of creation of this file + 2)
#else

#ifndef _LIBOR_AFFINE_FRAMEWORK_H
#define _LIBOR_AFFINE_FRAMEWORK_H

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

#include "pnl/pnl_vector.h"
#include "pnl/pnl_matrix.h"
#include "pnl/pnl_mathtools.h"

#include "math/read_market_zc/InitialYieldCurve.h"

#define phi_psi_t_v(t,v,LiborAffine,phi_i,psi_i) (((StructL
    iborAffine*)LiborAffine)->phi_psi)((((StructLiborAffine*)
    LiborAffine)->ModelParams), t, v, phi_i, psi_i)

typedef struct StructLiborAffine
{
  PnlVect *TimeDates; // Time Grid
  PnlVect *MartingaleParams;  // The parameters u_1,...,u_
    N, chosen in order to match the initial yield curve (at
    TimeDates)
  PnlVect *ModelParams; // Parameters of the driving proces
    s X, supposed to be affine.
  ZCMarketData *ZCMarket; // Structure that containts ini
    tial zero coupon bond.

  // Return the value of the 2 functions Phi and Psi in th
    e Moment Generating Function of an affine process.
  void (*phi_psi)(PnlVect *ModelParams, double t, dcomplex
    u, dcomplex *phi_i, dcomplex *psi_i);

  // Return maximum of the interval I_T, where Moment
    Generating Function is well defined.
```

```
  double (*MaxMgfArg)(PnlVect *ModelParams, double T);

} StructLiborAffine;


// Calibration of martingale parameters to match the initia
    l zero coupon curve.
void CreateStructLiborAffine(StructLiborAffine *LiborAffine
    , ZCMarketData *ZCMarket,
                             double T0, double TN, double
    Period, PnlVect *ModelParams,
                             void (*phi_psi)(PnlVect *,
    double, dcomplex, dcomplex *, dcomplex *),
                             double (*MaxMgfArg_cir1d)(PnlV
    ect *, double));

// Moment generating function of X(Ti) under the for forwar
    d measure P(T_fwd_meas)
dcomplex MomentGF_XTi_PTk(dcomplex v, double Ti, double T_
    fwd_meas, StructLiborAffine *LiborAffine);

// Moment generating function of X(Ti) under the for forwar
    d measure P(TN)
dcomplex MomentGF_XTi_PTN(dcomplex z, double Ti, StructLib
    orAffine *LiborAffine);

int indiceTimeLiborAffine(StructLiborAffine *LiborAffine,
    double s);
void FreeStructLiborAffine(StructLiborAffine *LiborAffine);

#endif
#endif
```

References