

Help

```

#include "variancegamma1d_pad.h"

int MOD_OPT(ChkMix)(Option *Opt, Model *Mod)
{
    TYPEOPT* ptOpt=( TYPEOPT*)(Opt->TypeOpt);
    TYPEMOD* ptMod=( TYPEMOD*)(Mod->TypeModel);
    int status=OK;

    if (ptOpt->Maturity.Val.V_DATE<=ptMod->T.Val.V_DATE)
    {
        Fprintf(TOSCREENANDFILE,"Current date greater than
        maturity!\n");
        status+=1;
    };
    if ((ptOpt->MinOrElse).Val.V_BOOL==MINIMUM)
    {
        if ((ptOpt->PathDep.Val.V_NUMFUNC_2)->Par[4].Val.V_
        PDOUBLE>ptMod->S0.Val.V_PDOUBLE)
        {
            Fprintf(TOSCREENANDFILE,"Minimum greater than spot!\n"
            );
            status+=1;
        };
    }
    if ((ptOpt->MinOrElse).Val.V_BOOL==MAXIMUM)
    {
        if ((ptOpt->PathDep.Val.V_NUMFUNC_2)->Par[4].Val.V_
        PDOUBLE<ptMod->S0.Val.V_PDOUBLE)
        {
            Fprintf(TOSCREENANDFILE,"Maximum lower than spot!\n");
            status+=1;
        };
    }
    return status;
}

//extern PricingMethod MET(AP_FixedAsian_FusaiMeucciVG);
extern PricingMethod MET(MC_VarianceGamma_Fixed);
extern PricingMethod MET(MC_VarianceGamma_Floating);

```

```
extern PricingMethod MET(MC_VarianceGamma_FixedAsian);
extern PricingMethod MET(MC_VarianceGamma_FloatingAsian);

PricingMethod *MOD_OPT(methods) []={
    //&MET(AP_FixedAsian_FusaiMeucciVG),
    &MET(MC_VarianceGamma_Fixed),
    &MET(MC_VarianceGamma_Floating),
    &MET(MC_VarianceGamma_FixedAsian),
    &MET(MC_VarianceGamma_FloatingAsian),
    NULL
};

DynamicTest* MOD_OPT(tests) []={
    NULL
};

Pricing MOD_OPT(pricing)={
    ID_MOD_OPT,
    MOD_OPT(methods),
    MOD_OPT(tests),
    MOD_OPT(ChkMix)
};
```

References