

[Help](#)

```
#include "
href../../mod/rstemperedstable1d/rstemperedstable1d_std/rstemperedstable1d_st

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;
    };

    return status;
}

extern PricingMethod MET(AP_CarrTS);
extern PricingMethod MET(AP_IAC);
extern PricingMethod MET(AP_ITKIN);
extern PricingMethod MET(AP_fastwhamer);
extern PricingMethod MET(AP_fastwhamerdigital);
//extern PricingMethod MET(AP_SPM_TSL);
extern PricingMethod MET(FD_ImpExp);
extern PricingMethod MET(FD_KLZ);
extern PricingMethod MET(FD_KLZ_EPUT);
extern PricingMethod MET(MC_TankovPoirot);

PricingMethod *MOD_OPT(methods) [] =
{
    &MET(AP_CarrTS),
    &MET(AP_IAC),
    &MET(AP_ITKIN),
    &MET(AP_fastwhamer),
    &MET(AP_fastwhamerdigital),
    //&MET(AP_SPM_TSL),
    &MET(FD_ImpExp),
```

```

    &MET(FD_KLZ),
    &MET(FD_KLZ_EPUT),
    &MET(MC_TankovPoirot),
    NULL
};

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

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

```