```
   Help
#include "pad.h"

extern char* path_sep;
extern char **error_msg;

int OPT(Get)(int user,Planning *pt_plan,Option *
    opt)
    {
    TYPEOPT* pt=(  TYPEOPT*)(opt->TypeOpt);

    (opt->Init)(opt);

    if (user==TOSCREEN)
        if  ((opt->Show)(user,pt_plan,opt))
            do
                {
                Fprintf(TOSCREEN,"_____________
    _____________Option:%s{n",opt->Name);

                ScanVar(pt_plan,user,&(pt->Maturi
    ty));
                GetParVar(pt_plan,user,(pt->PayOf
    f.Val.V_NUMFUNC_2)->Par);
                GetParVar(pt_plan,user,(pt->PathD
    ep.Val.V_NUMFUNC_2)->Par);
                }
                while ((opt->Show)(user,pt_plan,
    opt));

                return (opt->Show)(TOSCREENANDFI
    LE,pt_plan,opt);
    }

int OPT(Show)(int user,Planning *pt_plan,Option *
    opt)
    {
    TYPEOPT* pt=(TYPEOPT*)(opt->TypeOpt);

  (void)(opt->Init)(opt);
    Fprintf(user,"##Option:%s{n",opt->Name);
```

```
    PrintVar(pt_plan,user,&(pt->Maturity));
    ShowParVar(pt_plan,user,(pt->PayOff.Val.V_
    NUMFUNC_2)->Par);
    ShowParVar(pt_plan,user,(pt->PathDep.Val.V_
    NUMFUNC_2)->Par);

    return (opt->Check)(user,pt_plan,opt);
    }

int OPT(Check)(int user,Planning *pt_plan,Option
    *opt)
    {
    TYPEOPT* pt=(TYPEOPT*)(opt->TypeOpt);
    int status=OK;
    char helpfile[MAX_PATH_LEN]="";

  if ((strlen(opt->Name)+strlen(opt->ID)+strlen(
    "{{opt{{") +strlen("{{")
    +strlen("_doc.pdf"))>=MAX_PATH_LEN)
  {
    Fprintf(TOSCREEN,"%s{n",error_msg[PATH_TOO_
    LONG]);
        exit(WRONG);
  }

  strcpy(helpfile,path_sep);
  strcat(helpfile,"opt");
  strcat(helpfile,path_sep);

  strcat(helpfile,opt->ID);
  strcat(helpfile,path_sep);

  strcat(helpfile,opt->Name);
  strcat(helpfile,"_doc.pdf");


    status+=ChkVar(pt_plan,&(pt->Maturity));
    status+=((pt->PayOff.Val.V_NUMFUNC_2)->Check)
    (user,pt_plan,pt->PayOff.Val.V_NUMFUNC_2);
    status+=((pt->PathDep.Val.V_NUMFUNC_2)->Check
```

```
    )(user,pt_plan,pt->PathDep.Val.V_NUMFUNC_2);

    return Valid(user,status,helpfile);
    }

extern Option OPT(AsianCallFloatingEuro);
extern Option OPT(AsianPutFloatingEuro);
extern Option OPT(AsianPutFixedEuro);
extern Option OPT(AsianCallFixedEuro);
extern Option OPT(LookBackCallFixedEuro);
extern Option OPT(LookBackCallFloatingEuro);
extern Option OPT(LookBackPutFixedEuro);
extern Option OPT(LookBackPutFloatingEuro);
extern Option OPT(AsianCallFloatingAmer);
extern Option OPT(AsianPutFloatingAmer);
extern Option OPT(AsianPutFixedAmer);
extern Option OPT(AsianCallFixedAmer);
extern Option OPT(LookBackCallFixedAmer);
extern Option OPT(LookBackCallFloatingAmer);
extern Option OPT(LookBackPutFixedAmer);
extern Option OPT(LookBackPutFloatingAmer);

Option* OPT(family)[]={
        &OPT(LookBackCallFloatingEuro),
        &OPT(LookBackPutFloatingEuro),
        &OPT(LookBackPutFixedEuro),
        &OPT(LookBackCallFixedEuro),
        &OPT(AsianCallFixedEuro),
        &OPT(AsianPutFixedEuro),
        &OPT(AsianCallFloatingEuro),
        &OPT(AsianPutFloatingEuro),
        &OPT(LookBackCallFloatingAmer),
        &OPT(LookBackPutFloatingAmer),
        &OPT(LookBackPutFixedAmer),
        &OPT(LookBackCallFixedAmer),
        &OPT(AsianCallFixedAmer),
        &OPT(AsianPutFixedAmer),
        &OPT(AsianCallFloatingAmer),
        &OPT(AsianPutFloatingAmer),
        NULL
    };
```

References