

Appendix D. Function Prototypes for the *mathc90* Library

The following tabulation shows prototypes for all user-callable functions of the *mathc90* library, and some low-level functions as well. These prototypes are contained in the file *mathc90.h*.

```
void    ccoef(long,float[][2],float[][2]);
void    cdif(float[],float[],float[]);
void    cgam(float[],float[],float*,long);
void    cpolz(float[][2],long,float[][2],float*,long*);
void    cpro(float[],float[],float[]);
void    cquo(float[],float[],float[]);
void    csort(CHAR_INT,long,long,long,long);
void    csort1(char*[],long,long,long,long);
void    csortp(CHAR_INT,long,long,long,long,long[]);
void    csortq(CHAR_INT,long,long,long,long,long[]);
void    csqrtx(float[],float[]);
void    csum(float[],float[],float[]);
void    cwfz(float[],float[],long*);
void    daccum(double*,long,long,double*,long,long,long*,long,long*);
double  dasinh(double);
double  dacosh(double);
double  datanh(double);
double  dactnh(double);
double  dacsch(double);
double  dasech(double);
double  dhyps(double);
double  dhypc(double);
double  dhypb(double);
double  dhyper(double,double);
double  dasum(long,double[],long);
void    daxpy(long,double,double[],long,double[],long);
void    dbacc(double*,long,long,long*,long,long,long*,long*);
double  dbesj0(double);
double  dbesj1(double);
void    dbesjn(double,double,long,double[]);
void    dbesjq(double,double,double*,double*);
double  dbesy0(double);
double  dbesy1(double);
void    dbesyn(double,double,long,double[]);
void    dbi0k0(double,double*,double*,long,long*);
void    dbi1k1(double,double*,double*,long,long*);
double  dbinom(long,long);
void    dblse(double*,long,long,long,long,double[],double[][2],double,long,double,long*,double[],double*,long*,
double[],double[],double[],double[],double[],double[],long[],long[],double[],double[],double[]);
void    dbmp0(double,double*,double*);
void    dbmp1(double,double*,double*);
void    dbsol(long,double*,long,long,long,long,double[],long,double*,long*);
void    dc2bas(double,long,long,LOGICAL32*,double[],long,double[]);
void    dc2fit(double[],double[],double[],long,double[],long,double*,long,double[],double[],double*,long*);
void    dcdchi(double,double,double*,double*,long*);
double  dcdnml(double,double,double);
void    dcdpoi(long,double,double*,double*,long*);
void    dcft(double[],char*,long[],long,long*,double[]);
void    dchol(double*,long,long,double[],double*,double,long*);
double  dci(double);
double  dcin(double);
double  dcii(LOGICAL32,double);
void    dckder(long*,long,long,double[],double[],double*,long,double*,long*,long*,double*);
void    dcomqr(long,long,long,long,double*,double*,double[],long*);
void    dconcm(long,double[]);
```

```

void    dconmc(long,double[]);
void    dcopy(long,double[],long,double[],long);
double  dcos1(double);
double  dcoshm(double);
double  dcospx(double);
void    dcov2(double*,long,long,long[],double,long*);
void    dcov3(double*,long,long,double[],double,double[],long*);
void    dcpdrv(double[],long,double[],long*);
void    dcpint(double[],long,double[],long*);
double  dcplte(double);
double  dcpltk(double);
double  dcpval(double[],long,double);
double  dcsevl(double,double[],long);
double  dcshmm(double);
double  dcspxx(double);
void    ddas1(double*,double[],double[],long,long*,void*)(double*,double[],double[],double[],double[],long*,double*,
        long*,double[],long[]),long[],double*,double[],long*,double*,double[],double[],double[],long[],double[]);
void    ddasco(double*,long,long,double[],double[]);
void    ddasdb(long,long,double,double[],double[],long[],double[],long[],long,double[],double[]);
void    ddasf(double,double[],double[],double[],double[],long,double,long,double[],long[]);
void    ddasgh(double,double,double*,double,double*,double);
void    ddasin(double,double,double[],double[],long,long,double*,double[]);
void    ddasj(long,long*,double*,double[],double[],double[],double,double[],double[],double[],long[],double[],
        void*)(double*,double[],double[],double[],double[],long*,double*,long*,double[],long[]),long[],long*);
void    ddasls(void(*)(),long,double*,double[],double[],long[],double,double,double*,long*,long,long*,double[],long,
        long[],long);
void    ddaslv(long,long*,double*,double[],double[],double[],void*)(double*,double[],double[],double[],double[],long*,
        double*,long*,double[],long[]),long[],long[],double[]);
void    ddaslx(void(*)(),long,double*,double[],double[],double,long[],double[],double[],long*,double[],long,long[],long);
double  ddasnm(long,double[],double[],double[],long[]);
void    ddastp(double*,double[],double[],long,long*,void*)(double*,double[],double[],double[],double[],long*,double*,
        long*,double[],long[]),long[],double*,double[],long*,double*,double[],double[],double[],double[],long[],
        double[],double[],double[],double[],double[],double[],long*);
void    ddaswt(long,long[],double[],double[],double[],double[],double[],long[]);
double  ddot(long,double[],long,double[],long);
double  dei(double);
double  del(double);
void    delafi(double,double,double*,double*,long*);
void    delpii(double,double,double,double*,long*);
double  derf(double);
double  derfc(double);
double  derfce(double);
void    derfsp(LOGICAL32*,double*,double*,double*,double*);
double  derf1(double);
double  derfc1(double);
double  derfe1(double);
double  derfe2(double);
double  derfi(double);
double  derfci(double);
double  derfix(double,long);
void    derm1(char*,long,long,char*,char*,double,byte);
void    derv1(char*,double,byte);
void    devbh(double*,long,long,long*,long*,long[],double[]);
void    devun(double*,long,long,double[],double[],long[]);
void    devvun(double*,long,long,double[],double[],double*,long[],double[]);
void    dcdiv(double,double,double,double,double*,double*);
void    dfft(double[],double[],double[]);
void    dfmin(double*,double*,long*,double);
double  dfrenc(double);

```

```

double  dfrenf(double);
double  dfreng(double);
double  dfrens(double);
double  dfren1(long,double);
double  dgam1(double);
void    dgame(double,double,double*,double*,long*);
void    dgamik(double,double,double,long);
void    dgame(double*);
void    dgamib(void);
double  dgamma(double);
void    dgbfa(double*,long,long,long,long,long[],long*);
void    dgbsl(double*,long,long,long,long,long[],double[],long);
void    dgeco(double*,long,long,long[],double*,double[]);
void    dged(double*,long,long,long[],double[]);
void    dgefa(double*,long,long,long[],long*);
void    dgefs(double*,long,long,double*,long,long,long[],long*);
void    dgefsc(double*,long,long,double*,long,long,long[],double*,double[]);
void    dgei(double*,long,long,long[],double[]);
void    dgemv(byte,long,long,double,double*,long,double[],long,double,double[],long);
void    dgesl(double*,long,long,long[],double[],long);
void    dgesld(double*,long,long,long[],double[]);
void    dgeslt(double*,long,long,long[],double[]);
void    dgr17(double,double,double*);
void    dgr29(double,double,double*);
void    dherql(double*,double*,long,long,double[],double*,double*,double[],long*);
void    dhfti(double*,long,long,long,double*,long,long,double,long*,double[],double[],long[]);
double  dhint(double,long,long,double[],double[],double[]);
void    dhfcc(long,long,long,long,double[],double*,double[],long,long);
void    dhfgen(long,long,long,long,double[],long,LOGICAL32,double*,double[],long,long,LOGICAL32);
void    dilup(double,double*,long,double[],double[],long,long*,long[],double[]);
void    dilupm(long,double[],double*,long[],double[],double[],long[],long[],long[],double[]);
void    dilupmd(long,double[],double*,long[],double[],double[],long[],long[],long[],double[]);
void    dimql(double*,long,long,double[],double[],long*);
void    dinit(double[],long,double,long*);
void    dint1(double,double,double*,double[],long[]);
void    dinta(double*,double[],long[]);
void    dintdl(double[]);
void    dintdu(void);
void    dintf(double*,double[],long[]);
void    dintm(long,double*,double[],long,long[]);
void    dintma(double*,double[],long[]);
void    dintns(long);
void    dintn(long,double[]);
void    dintop(long[],double[]);
double  dintsm(double);
void    diva(double[],double[],double[],long[],long,void*)(double[],double[],double[],long[]),void*(double[],
    double[],double[],long[]),long,long,long,long,long[]);
void    divaa(double[],double[],double[],long[],void*)(double[],double[],double[],long[]),void*(double[],double[],
    double[],long[]);
void    divabu(double[],long[]);
void    divaco(long[],double[]);
void    divacr(double[],double[],long[],double[],long[]);
void    divahc(void);
void    divain(double[],double[],double[],long[]);
void    divaop(long[],double[]);
void    divapr(double[],double[],double[],long[]);
void    divadb(long,double[],double[],double[],long[],char*);
void    divag(double[],double[],double[],long[],long*,long*,double[],double[]);
void    divset(long,long[],long,long,double[]);

```

```

void    dv7dff(long,long,double[]);
double  dr7mdc(long);
double  dl7svn(long,double[],double[],double[]);
void    dq7apl(long,long,long,double*,double[],long);
double  dv2nrm(long,double[]);
double  dd7tpr(long,double[],double[]);
void    dd7upd(double[],double*,long[],long,long,long,long,long,long,long,double[]);
void    dq7rad(long,long,long,double[],LOGICAL32,double[],double*,double[]);
void    dparck(long,double[],long[],long,long,long,double[]);
void    ds7lvm(long,double[],double[],double[]);
void    ds7lup(double[],double,long,double,double[],double[],double[],double[],double*,double[]);
void    dl7mst(double[],double[],long,long[],long*,long,double[],double[],double[],double[],double[]);
void    dg7qts(double[],double[],double[],long*,double[],long,double[],double[],double[]);
void    dl7ivm(long,double[],double[],double[]);
double  dl7svx(long,double[],double[],double[]);
void    da7sst(long[],long,long,double[]);
void    ditsum(double[],double[],long[],long,long,long,double[],double[]);
void    dl7itv(long,double[],double[],double[]);
void    dl7sqr(long,double[],double[]);
void    dl7srt(long,long,double[],double[],long*);
void    dl7tvm(long,double[],double[],double[]);
void    dl7vml(long,double[],double[],double[]);
double  drldst(long,double[],double[],double[]);
void    dv2axy(long,double[],double,double[],double[]);
void    dv7cpy(long,double[],double[]);
void    dv7scl(long,double[],double,double[]);
void    dv7scp(long,double[],double);
void    djacg(long*,long,long,double[],double[],double*,long,double[],double[],long[],double[],long,long[],long);
void    dlasum(double,long,double[],double*);
void    dlesum(double,long,double[],double*);
double  dlgamma(double);
double  drat1(double);
double  dlnrel(double);
void    dmatp(double*,long,long,long,char*);
void    dmatpr(double*,long,long,long,char*,long,long,long);
void    dmess(long[],CHAR_INT,long[],double[]);
void    void(*) (long,double[],double*,double[],LOGICAL32*),long,long,long,double*,long,double[],double[],double[],
        double[],double,long,long,long[],long,double[],long);
void    void(*) (long,double[],double*,double[],LOGICAL32*),long,long,long,double*,long,double[],double[],double[],
        double[],double,long[],long*,double[],long,long,long*,long*,long*,double[],double[],double[],
        double[],double[],double*,double[],double[],double[],double[],double[],double[],double*,double[]);
void    dmlc03(long,long,long,double*,long,double[],double[],long[],long*,long*,double[],double,double[],double[]);
void    dmlc04(long,long,double*,long,double[],double[],double[],double[],long[],long*,long*,double[],long*,double[],
        double[],double[],double[],double,double*,long,long*,long,double[],double[],double[],double[],
        double[],double[],double[],double[]);
void    void(*) (long,double[],double*,double[],LOGICAL32*),long,long,double*,long,double[],double[],double[],
        double[],double,long[],long*,double[],long,long,long,long*,double[],double[],double[],
        double[],double,double*,double,long,long,long*,long*,long*,long,double[],double*,double[],
        double[],double[],double[],double[],double[],double*,long*,LOGICAL32*,double[]);
void    dmlc06(long,long,double*,long,double[],double[],double[],double[],long[],long*,double[],double[],double[],
        double[],double[],double[],double[],double[],double,double,double*,double*,long,long*,long,long*,
        double[],double[],double[],double[],double[]);
void    dmlc07(long,long,double*,long,long[],long*,double[],double[],double[],double[],double[],double,
        double*,long,long,double[],double[],double[],double[]);
void    dmlc08(long,long,double*,long,long[],long*,double[],double[],double[],double[],double[],double[],double,
        double*,long,long,double[],double[],double[]);
void    void(*) (long,double[],double*,double[],LOGICAL32*),long,double[],double[],double[],double[],double[],
        double,double,double,double*,double*,long*,long,double[],double[],double[],long*,double[]);
void    dmlc10(long,long,double*,long,long[],long*,double[],double[],double[],double,long,double[],double[],double[]);

```

```

void      dmlc11(long,long,double*,long,double[],double[],double[],double[],long[],long*,long*,double[],double[],
              double[],double,double,long);
void      dmlc12(long,long,double*,long,long[],long*,double[],double[],double,long,double[],double[]);
void      dmlc13(long,long,double*,long,double[],double[],double[],double[],long[],long,double[],double,double*,long);
void      dmlc14(long,long,double*,long,long[],long*,double[],double[],double,long);
void      dmlc15(long,long,double[],double[],double[],long[],long*,long*,double[],double[],double[],double*);
void      dmlc16(long,long,double*,long,long[],long,double[],double[],double[],double[],double[],double[],double*,long,
              double*,double[],double[]);
void      dmlc17(long,long,double[],double[],double[],double[],double,double*);
void      dmlc18(long,long,double*,long,long[],double[],double[],double*,double*,long,long*,long,long*);
void      dmlc19(long,double[],long,double[],double[],double[],double[],double[],double*);
void      void*(long,double[],double*,double[],LOGICAL32*),long,double[],double,double[],double[],double[],long*,
              long*,double[]);
void      dmlc21(long,LOGICAL32,long,long,long,double,double,double[],double[],long,long[],double[],double);
void      dmpdrv(double[],long,double[],long*);
void      dmpint(double[],long,double[],long*);
double    dmpval(double[],long,double);
void      dnlafe(long,long,double[],double[][2],void*(long,long,double[],long*,double[]),long[],long,long,double[]);
void      dnlafu(long,long,double[],void*(long,long,double[],long*,double[]),long[],long,long,double[]);
void      dnlagb(long,long,double[],double[][2],void*(long,long,double[],long*,double[]),void*(long,long,double[],
              long*,double[]),long[],long,long,double[]);
void      dnlagu(long,long,double[],void*(long,long,double[],long*,double[]),void*(long,long,double[],long*,
              double[]),long[],long,long,double[]);
void      dnlsfb(long,long,long,double[],double[][2],double[],double[],void*(long,long,long,double[],long*,double[]),
              long*,long,long[],long,long,double[]);
void      dnlsfu(long,long,long,double[],double[],double[],void*(long,long,long,double[],long*,double[]),long*,long,
              long[],long,long,double[]);
void      dnlsfb(long,long,long,double[],double[][2],double[],double[],void*(long,long,long,double[],long*,double[]),
              void*(long,long,long,double[],long*,double[]),long*,long,long[],long,long,double[]);
void      dnlsfu(long,long,long,double[],double[],double[],void*(long,long,long,double[],long*,double[]),void*(long,
              long,long,double[],long*,double[]),long*,long,long[],long,long,double[]);
void      void*(long,double[],double[],double*,long*),long,double[],double[],double[],long[],double[],long);
void      void*(long,double[],double[],double*,long*),long,double[],double[],double,long*,long*,long*,long,
              LOGICAL32,long,LOGICAL32,long,long,double,double,LOGICAL32,double*,double[],double[],
              double[],double[],double[],double[],double[],double*,double[]);
void      void*(long,double[],double[],double*,long*),long,double[],double[],double*,long,long*,long,long,double,
              double[],double[]);
void      dnqaq(long,long,double*,long,double[],double[]);
void      dnqdog(long,double[],long,double[],double[],double,double[],LOGICAL32*,double[],double[],LOGICAL32,double[]);
void      dnqqfm(long,long,double*,long,double[]);
void      dnqqrf(long,long,double*,long,LOGICAL32,long[],long,double[],double[],double[]);
void      dnqupd(long,long,double[],long,double[],double[],double[],LOGICAL32*);
double    dnrm2(long,double[],long);
void      dpfit(long,double[],double[],double[],long,LOGICAL32,LOGICAL32,LOGICAL32,double[],long*,double*,double*);
void      dplot(double,double,double[],long,double[],double[],STRING);
void      dplota(long);
void      dplote(long,double[],STRING);
void      dplotf(long,double[],double[],double[]);
void      dplotn(double,long,double[]);
void      dplott(long,double[]);
void      dplotr(double[],long,long,long);
void      dplot0(void);
void      dplot1(void);
void      dplot2(double,double,double,double);
void      dplot4(double,double,char*,char*);
void      dplot5(double,double,double,double);
void      dplot6(double,double,double,double,double);
void      dplot7(long*,long[],double[]);
void      dplot8(double,double,double,double,double,double,long,double);

```

```

void    dplot9(void);
void    dplot1(long,double[],double[]);
void    dplots(double[],long);
void    dpolz(double[],long,double[],double*,long*);
void    dpolz2(double[],double[]);
double  dppnml(double,double,double);
double  dpquad(long,long,double[],double*,double,double);
void    dprpl(double,byte,byte[],long,double,double,LOGICAL32);
void    dprpl1(double[],double[],long,char*,char*,char*,long,long,byte[],long*);
void    dprpl2(double*,long,long,long[],long[],long[],byte[],char*,char*,char*,long,long,byte[],long*);
void    dprpl3(double,double,double,double,double*,double*,double*,double*,long*,long*,long*,long*,char*,char*,char*,
        long,long,byte[],long*);
void    dprpl4(double,double,double*,double*,long*,long*,byte[6],long*,long*);
void    dprpl5(double,double,long,byte[6],long,long,long,long,byte[]);
void    dprtsv(double*,long,long,long,CHAR_INT,long,long,long);
double  dpsi(double);
void    dpsik(double,double,long);
void    dpsie(double*,long*);
void    dpsib(void);
double  dpval(long,long,double[],double*,double,long);
void    dq7rfh(long*,long[],long,long,long,long,double*,double[],long,double[]);
void    ds7cpr(double[],long[],long,long);
void    dv7prm(long,long[],double[]);
void    dv7swp(long,double[],double[]);
void    dqrbd(long*,double[],double[],long,double*,long,long,double*,long,long);
double  drane(double);
double  drang(void);
void    drangv(double*,long,long,double[],double[],LOGICAL32*,long*);
double  dranr(double);
double  dranu(void);
double  drcomp(double,double);
void    drcval(double,double,double*,long*);
void    drdval(double,double,double,double*,long*);
double  drexp(double);
void    drft(double[],byte,long[],long,long*,double[]);
void    drft1(double[],byte,long,long*,double[]);
void    drfval(double,double,double,double*,long*);
void    drfvlx(double,double,double,double*);
void    drjval(double,double,double,double,double*,long*);
double  drlog(double);
double  drlog1(double);
double  drlog2(double);
void    drn2g(double[],double*,long[],long,long,long,long,long*,long*,long,double[],double[],double[],double[]);
void    dg7lit(double[],double[],long[],long,long,long,long,double[],double[],double[]);
void    dn2lrd(double*,long[],double[],long,long,long,long,long,long,double[],double[],double[]);
void    dc7vfm(long[],double[],long,long,long,long,long,double[]);
void    df7hes(double[],double[],long*,long[],long,long,long,double[],double[]);
void    dn2cvp(long[],long,long,long,double[]);
void    dn2rdp(long[],long,long,double[]);
void    do7prd(long,long,long,double[],double[],double*,double*);
void    dl7nvr(long,double[],double[]);
void    dl7tsq(long,double[],double[]);
void    drn2gb(double[][2],double[],double*,long[],long,long,long,long,long*,long*,long,double[],double[],double[],double[]);
void    dg7itb(double[][2],double[],double[],long[],long,long,long,long,double[],double[],double[]);
void    dr7tvm(long,long,double[],double[],double*,double[]);
void    df7dhh(double[][2],double[],double[],long*,long[],long,long,long,double[],double[]);
double  dh2rfg(double,double,double*,double*,double*);
void    dh2rfa(long,double[],double[],double,double,double);
void    dg7qsb(double[][2],double[],double[],double[],long[],long[],long[],long*,double[],long,long,long*,long,double*,

```

```

        double[],double[],double[],double[],double[],double[]);
void dl7msb(double[][2],double[],double[],long,long[],long[],long[],long*,double[],long,long,long*,long,double[],
        double[],double*,double[],double[],double[],double[],double[],double[]);
void ds7bqn(double[][2],double[],double[],long[],long[],long[],long*,double[],long,long*,long,long*,double[],
        double[],double[],double[],double[],double[],double[]);
void dq7rsh(long,long,LOGICAL32,double[],double[],double[]);
void dv7vmp(long,double[],double[],double[],long);
void dv7ipr(long,long[],double[]);
void dv7shf(long,long,double[]);
void ds7ipr(long,long[],double[]);
void dd7mlp(long,double[],double[],double[],long);
void ds7dmp(long,double[],double[],double[],long);
void drnsg(double*,double[],double[],double*,long[][2],long[],long,long,long,long,long,long,long,long,double[],double[]);
void drnsgb(double*,double[],double[][2],double[],double*,long[][2],long[],long,long,long,long,long,long,long,long,
        double[],double[]);
void drot(long,double[],long,double[],long,double,double);
void drotg(double*,double*,double*,double*);
void drotm(long,double[],long,double[],long,double[]);
void drotmg(double*,double*,double*,double,double[]);
void dsbasd(long,long,double[],double,long,double[]);
void dsbasi(long,long,double[],double,double,long*,long*,double[]);
void dscal(long,double,double[],long);
void dsdif(long,long,double[],double[],long,double*);
double dsdot(long,float[],long,float[],long);
void dsfind(double[],long,long,double,long*,long*);
void dsfit(double[],double[],double[],long,long,long,double[],double[],double*,long*,long,double*);
void dsfite(byte[][5],double[],double[],double[],long,long,double[],double[],double*,long[],long[],double[]);
double dsi(double);
double dsin1(double);
double dsinhm(double);
double dsinpx(double);
double dsnpxx(double);
void dsort(double[],long,long);
void dsortp(double[],long,long,long[]);
void dsortq(double[],long,long,long[]);
void dspge(long,long[],long[],double[],double[],double[]);
double dsquad(long,long,double[],double[],double,double);
void dstat1(double[],long,double[],long[],long,double,double);
void dstat2(double[],long[],long,double,double);
void dstop(long,long,double[],double[],double*,long*,double[],double*);
void dsva(double*,long,long,long,long,double[],double[],long[],CHAR_INT,long,double[],double[]);
double dsval(long,long,double[],double[],double,long);
void dsvala(long,long,double[],long,double*,double,double[]);
void dsvdrs(double*,long,long,long,double*,long,long,double[],double[]);
void dswap(long,double[],long,double[],long);
void dsymql(double*,long,long,double[],double[],long*);
void dtcst(double[],char*,char*,long[],long,long*,double[]);
void dtgc0(double[][3],double*,LOGICAL32,double[]);
void dtgc1(LOGICAL32,double[][3],double*,LOGICAL32,double[]);
void dtgext(double[],double[],double[],double[][2],long[],long[][4],long,long,double[],long,long,double*,
        LOGICAL32,double[]);
void dtgqs(double[],long[],double[],double[],double[][3]);
void dtgfi(double[],double[],double[],double[][2],long[],long,long[][4-(1+1)],long,long,double[],double*,
        LOGICAL32,double[],long*,double[]);
void dtgfind(double[],double[],long[],long,double[],long*,long[],double[][3],long*);
void dtggrd(double[],double[],long,long[],double[],long[],long,long[][4],long,long*,long[]);
void dtgupd(long,long,long[],long[],long,long);
double dtgang(double,double,double);
void dtgadj(long,long,double[],double[],long,long[],long,long[][4],long,long,LOGICAL32*);

```

```

void dtgpd(double[],double[],double[],double[][2],long,long[],long,long[]);
void dtgmor(double[],double[],double[],long,long[],double[][21]);
void dtgls(double[][21],long,long,LOGICAL32*,long,double*,double*);
void dtgprg(double[],double[],long,long[],long[][4],long,long);
void dtgrec(double[],double[],double[],double[][2],long,long[],long,long[][4],long,double[],long,long,double,
double*,long,long,long,LOGICAL32,double*);
void dtgset(long,long,long,long,long,long,long,long[],long);
void dtgget(long,long[],long[]);
void dtgput(long,long[],long[],long);
void dtgsiz(long,long*);
void dtrc2c(double[],long,double[],double[],double[]);
void dusetn(long,long,long);
void dugetn(long*,long*,long*,long*,long*);
void duset(double,long,double[]);
void dupro(double[],double[],double[]);
void duquo(double[],double[],double[]);
void dusum(double[],double[],double[]);
void dudif(double[],double[],double[]);
void dusum1(double,double[],double[]);
void dudif1(double,double[],double[]);
void dupro1(double,double[],double[]);
void duquo1(double,double[],double[]);
void dusqrt(double[],double[]);
void duexp(double[],double[]);
void dulog(double[],double[]);
void dupwri(long,double[],double[]);
void dusin(double[],double[]);
void ducos(double[],double[]);
void dusinh(double[],double[]);
void ducosh(double[],double[]);
void duatan(double[],double[]);
void duatn2(double[],double[],double[]);
void duasin(double[],double[]);
void duacos(double[],double[]);
void duacs(LOGICAL32,double[],double[]);
void dutan(double[],double[]);
void dutanh(double[],double[]);
void durev(double*,double*,long,double*,long[],double*);
void dvecp(double[],long,char*);
void dvecpr(double[],long,char*,long,long,long);
void dwatan(long,double[],double[]);
void dwasin(long,double[],double[]);
void dwacos(long,double[],double[]);
void dwacsi(long,double[],double[],LOGICAL32);
void dwatn2(long,double[],double[],double[]);
void dwsum(long,double[],double[],double[]);
void dwdif(long,double[],double[],double[]);
void dwsqrt(long,double[],double[]);
void dwexp(long,double[],double[]);
void dwsin(long,double[],double[]);
void dwcos(long,double[],double[]);
void dwtan(long,double[],double[]);
void dwsinh(long,double[],double[]);
void dwcosh(long,double[],double[]);
void dwtanh(long,double[],double[]);
void dwset(long,double,double,double[]);
void dwsum1(long,double,double[],double[]);
void dwdif1(long,double,double[],double[]);
void dwpro1(long,double,double[],double[]);

```



```

void      dwquo1(long,double,double[],double[]);
void      dwlog(long,double[],double[]);
void      dwpwri(long,long,double[],double[]);
void      dwchn(long,double[],double[]);
void      dwrchn(long,double[],double[]);
void      dwpro(long,double[],double[],double[]);
void      dwquo(long,double[],double[],double[]);
void      dpascl(long,double[]);
double    dxparg(long);
void      dxrk8(double[],double[],double[],long[],double[],double[]);
void      dxrk8a(double[],double[],double[],long[],double[],double[]);
void      dxrk8i(double,double[],long[],double[]);
void      dxrk8n(long[],double[],double[],double[],double[],double[],double[]);
double    dxrk8x(double,double,double,double);
void      dxrk8f(double*,double[],double[],long[]);
void      dxrk8g(double[],double[],double[],long[]);
void      dxrk8o(double[],double[],long[],double[]);
double    dzabs(double[]);
void      dzero(double*,double*,double*,double*,long*,double);
void      erfin(void);
void      ermor(char*,byte);
void      ermsg(char*,long,long,char*,byte);
void      ermset(long);
void      void(*) (long,long,long,long*),long,long[],long,long*);
void      long(*) (long,long),long,long[]);
void      i7copy(long,long[],long[]);
void      i7pnvr(long,long[],long[]);
void      i7shft(long,long,long[]);
long      idamax(long,double[],long);
long      idranp(double);
void      idsm(long,long,long,long[],long[],long[],long*,long*,long*,long[],long[],long[],long,long[]);
void      i7rtdt(long,long,long[],long[],long[],long[]);
void      is7etr(long,long,long[],long[],long[],long[],long[]);
void      id7egr(long,long[],long[],long[],long[],long[],long[],long[]);
void      m7slo(long,long[],long[],long[],long[],long[],long*,long[],long[],long[],long[],long[]);
void      m7seq(long,long[],long[],long[],long[],long[],long*,long[],long[]);
void      i7do(long,long,long[],long[],long[],long[],long[],long*,long[],long[],long[],long[],long[]);
void      n7msrt(long,long,long[],long,long[],long[],long[]);
void      idsta1(long[],long,long[],double[],long[],long,long);
void      idsta2(long[],double[],long[],long,long);
void      ierm1(char*,long,long,char*,char*,long,byte);
void      ierv1(char*,long,byte);
void      imatp(long*,long,long,long,char*);
void      imatpr(long*,long,long,long,char*,long,long);
void      long(*) (long,long),long,long[],long*);
void      void(*) (long,long,long,long*),long,long[],long*);
long      isamax(long,float[],long);
void      isort(long[],long,long);
void      isortp(long[],long,long,long[]);
void      isortq(long[],long,long,long[]);
long      isranp(float);
void      issta1(long[],long,long[],float[],long[],long,long);
void      issta2(long[],float[],long[],long,long);
void      ivecp(long[],long,char*);
void      ivecpr(long[],long,char*,long,long);
LOGICAL32 lsame(byte,byte);
void      mess(long[],CHAR_INT,long[]);
void      messfd(long[]);
void      messfi(void);

```

```

void messmh(CHAR_INT); void messpr(void);
void messft(long[],char*);
void optchk(long[],long[],char*);
void pvec(long[],long);
void ran1(void);
void ran0(void);
void ranput(long[]);
void rn1(void);
void ransiz(long*);
void rnput(long[]);
void ranget(long[]);
void rn2(long*);
void sranua(float[],long);
void dranua(double[],long);
void sranus(float[],long,float,float);
void dranus(double[],long,double,double);
void ranmod(void);
void saccum(float*,long,long,float*,long,long,long*,long,long*);
float sasinh(float);
float sacosh(float);
float satanh(float);
float sactnh(float);
float sacsch(float);
float sasech(float);
float shyps(float);
float shyph(float);
float shyper(float,float);
float sasum(long,float[],long);
void saxpy(long,float,float[],long,float[],long);
void sbacc(float*,long,long,long*,long,long,long*,long*);
float sbesj0(float);
float sbesj1(float);
void sbesjn(float,float,long,float[]);
void sbespq(float,float,float*,float*);
float sbesy0(float);
float sbesy1(float);
void sbesyn(float,float,long,float[]);
void sbi0k0(float,float*,float*,long,long*);
void sbi1k1(float,float*,float*,long,long*);
float sbinom(long,long);
void sblse(float*,long,long,long,long,float[],float[][2],float,long,float,long*,float[],float*,long*,float[],
float[],float[],float[],float[],float[],long[],long[],float[],float[],float[]);
void sbmp0(float,float*,float*);
void sbmp1(float,float*,float*);
void sbisol(long,float*,long,long,long,long,float[],long,float*,long*);
void sc2bas(float,long,long,LOGICAL32*,float[],long,float[]);
void sc2fit(float[],float[],float[],long,float[],long,float*,long,float[],float[],float*,long*);
float scabs(float[]);
void scdchi(float,float,float*,float*,long*);
float scdnml(float,float,float);
void scdpoi(long,float,float*,float*,long*);
void scft(float[],char*,long[],long,long*,float[]);
void schol(float*,long,long,float[],float*,float*,long*);
float sci(float);
float scin(float);
float scii(LOGICAL32,float);
void sckder(long*,long,long,float[],float[],float*,long,float*,long*,long*,float*);
void scomqr(long,long,long,long,float*,float*,float[],long*);

```

```

void sconcm(long,float[]);
void sconmc(long,float[]);
void scopy(long,float[],long,float[],long);
float scos1(float);
float scoshm(float);
float scospx(float);
void scov2(float*,long,long,long[],float,long*);
void scov3(float*,long,long,float[],float,float[],long*);
void scpdrv(float[],long,float[],long*);
void scpint(float[],long,float[],long*);
float scplte(float);
float scpltk(float);
float scpval(float[],long,float);
float scsevl(float,float[],long);
float scshmm(float);
float scspxx(float);
void sdas1(float*,float[],float[],long,long*,void*)(float*,float[],float[],float[],float[],long*,float*,long*,
float[],long[]),long[],float*,float[],long*,float*,float[],float[],float[],long[],float[]);
void sdasco(float*,long,long,float[],float[]);
void sdasdb(long,long,float,float[],float[],long[],float[],long[],long,float[],float[]);
void sdasf(float,float[],float[],float[],float[],long,float,long,float[],long[]);
void sdasgh(float,float,float*,float,float*,float);
void sdasin(float,float,float[],float[],long,long,float*,float[]);
void sdasj(long,long*,float*,float[],float[],float[],float,float[],float[],float[],long[],float[],void*)(float*,
float[],float[],float[],float[],long*,float*,long*,float[],long[]),long[],long*);
void sdasls(void(*)(),long,float*,float[],float[],long[],float,float,float*,long*,long*,float[],long,long[],long);
void sdaslv(long,long*,float*,float[],float[],float[],void*)(float*,float[],float[],float[],float[],long*,float*,
long*,float[],long[]),long[],long[],float[]);
void sdaslx(void(*)(),long,float*,float[],float[],float,long[],float[],float[],long*,float[],long,long[],long);
float sdasnm(long,float[],float[],float[],long[]);
void sdastp(float*,float[],float[],long,long*,void*)(float*,float[],float[],float[],float[],long*,float*,long*,
float[],long[]),long[],float*,float[],long*,float*,float[],float[],float[],long[],float[],float[],
float[],float[],float[],float[],long*);
void sdaswt(long,long[],float[],float[],float[],float[],float[],long[]);
float sdot(long,float[],long,float[],long);
float sdsdot(long,float,float[],long,float[],long);
float sei(float);
float sel(float);
void selefi(float,float,float*,float*,long*);
void selpii(float,float,float,float*,long*);
float serf(float);
float serfc(float);
float serfce(float);
void serfsp(LOGICAL32*,float*,float*,float*,float*);
float serf1(float);
float serfc1(float);
float serfel(float);
float serfe2(float);
float serfi(float);
float serfci(float);
float serfix(float,long);
void serm1(char*,long,long,char*,char*,float,byte);
void serv1(char*,float,byte);
void sevbh(float*,long,long,long*,long*,long[],float[]);
void sevun(float*,long,long,float[],float[],long[]);
void sevvn(float*,long,long,float[],float[],float*,long[],float[]);
void sctdiv(float,float,float,float,float*,float*);
void sfft(float[],float[],float[]);
void sfmin(float*,float*,long*,float);

```

```

float  sfrenc(float);
float  sfrenf(float);
float  sfreng(float);
float  sfrens(float);
float  sfrenl(long,float);
float  sgam1(float);
void   sgami(float,float,float*,float*,long*);
void   sgamik(float,float,float,long);
void   sgamie(float*);
void   sgamib(void);
float  sgamma(float);
void   sgbfa(float*,long,long,long,long,long[],long*);
void   sgbsl(float*,long,long,long,long,long[],float[],long);
void   sgeco(float*,long,long,long[],float*,float[]);
void   sged(float*,long,long,long[],float[]);
void   sgefa(float*,long,long,long[],long*);
void   sgefs(float*,long,long,float*,long,long,long[],long*);
void   sgefsc(float*,long,long,float*,long,long,long[],float*,float[]);
void   sgei(float*,long,long,long[],float[]);
void   sgemv(byte,long,long,float,float*,long,float[],long,float,float[],long);
void   sgesl(float*,long,long,long[],float[],long);
void   sgesld(float*,long,long,long[],float[]);
void   sgeslt(float*,long,long,long[],float[]);
void   sgr17(float,float,float*);
void   sgr29(float,float,float*);
void   sherql(float*,float*,long,long,float[],float*,float*,float[],long*);
void   shfti(float*,long,long,long,float*,long,long,float,long*,float[],float[],long[]);
float  shint(float,long,long,float[],float[],float[]);
void   shtcc(long,long,long,long,float[],float*,float[],long,long);
void   shtgen(long,long,long,long,float[],long,LOGICAL32,float*,float[],long,long,LOGICAL32);
void   silup(float,float*,long,float[],float[],long,long*,long[],float[]);
void   silupm(long,float[],float*,long[],float[],float[],long[],long[],long[],float[]);
void   silupmd(long,float[],float,long[],float[],float[],long[],long[],long[],float[]);
void   simql(float*,long,long,float[],float[],long*);
void   sinits(float[],long,float,long*);
void   sint1(float,float,float*,float[],long[]);
void   sinta(float*,float[],long[]);
void   sintdl(float[]);
void   sintdu(void);
void   sintf(float*,float[],long[]);
void   sintm(long,float*,float[],long,long[]);
void   sintma(float*,float[],long[]);
void   sintns(long);
void   sinto(long,float[]);
void   sintop(long[],float[]);
float  sintsm(float);
void   siva(float[],float[],float[],long[],long,void*)(float[],float[],float[],long[]),void*(float[],float[],
    float[],long[]),long,long,long,long,long[]);
void   sivaa(float[],float[],float[],long[],void*)(float[],float[],float[],long[]),void*(float[],float[],float[],long[]);
void   sivabu(float[],long[]);
void   sivaco(long[],float[]);
void   sivacr(float[],float[],long[],float[],long[]);
void   sivahc(void);
void   sivain(float[],float[],float[],long[]);
void   sivaop(long[],float[]);
void   sivapr(float[],float[],float[],long[]);
void   sivadb(long,float[],float[],float[],long[],char*);
void   sivag(float[],float[],float[],long[],long*,long*,float[],float[]);
void   sivset(long,long[],long,long,float[]);

```



```

void smlc11(long,long,float*,long,float[],float[],float[],float[],long[],long*,long*,float[],float[],float[],float,
float,long);
void smlc12(long,long,float*,long,long[],long*,float[],float[],float,long,float[],float[]);
void smlc13(long,long,float*,long,float[],float[],float[],float[],long[],long,float[],float,float*,long);
void smlc14(long,long,float*,long,long[],long*,float[],float[],float,long);
void smlc15(long,long,float[],float[],float[],long[],long*,long*,float[],float[],float[],float*);
void smlc16(long,long,float*,long,long[],long,float[],float[],float[],float[],float[],float[],float*,long,float*,
float[],float[]);
void smlc17(long,long,float[],float[],float[],float[],float,float*);
void smlc18(long,long,float*,long,long[],float[],float[],float*,float*,long,long*,long,long*);
void smlc19(long,float[],long,float[],float[],float[],float[],float*);
void void*(long,float[],float*,float[],LOGICAL32*),long,float[],float,float[],float[],float[],long*,long*,float[]);
void smlc21(long,LOGICAL32,long,long,long,float,float,float[],float[],long,long[],float[],float[],float);
void smpdrv(float[],long,float[],long*);
void smpint(float[],long,float[],long*);
float smpval(float[],long,float);
void snlafb(long,long,float[],float[][2],void*(long,long,float[],long*,float[]),long[],long,long,float[]);
void snlafu(long,long,float[],void*(long,long,float[],long*,float[]),long[],long,long,float[]);
void snlagb(long,long,float[],float[][2],void*(long,long,float[],long*,float[]),void*(long,long,float[],long*,
float[]),long[],long,long,float[]);
void snlagu(long,long,float[],void*(long,long,float[],long*,float[]),void*(long,long,float[],long*,float[]),
long[],long,long,float[]);
void snlsfb(long,long,long,float[],float[][2],float[],float[],void*(long,long,long,float[],long*,float[]),long*,
long,long[],long,long,float[]);
void snlsfu(long,long,long,float[],float[],float[],void*(long,long,long,float[],long*,float[]),long*,long,long[],
long,long,float[]);
void snlsgb(long,long,long,float[],float[][2],float[],float[],void*(long,long,long,float[],long*,float[]),
void*(long,long,long,float[],long*,float[]),long*,long,long[],long,long,float[]);
void snlsgu(long,long,long,float[],float[],float[],void*(long,long,long,float[],long*,float[]),void*(long,long,
long,float[],long*,float[]),long*,long,long[],long,long,float[]);
void void*(long,float[],float[],float*,long*),long,float[],float[],float,long*,long*,long*,long,LOGICAL32,
long,LOGICAL32,long,long,float,float,LOGICAL32,float*,float[],float[],float[],float[]);
void void*(long,float[],float[],float*,long*),long,float[],float[],float*,long,long*,long,long,float,float[],float[]);
void snqaq(long,long,float*,long,float[],float[]);
void snqdog(long,float[],long,float[],float[],float,float[],LOGICAL32*,float[],float[],LOGICAL32,float[]);
void snqqfm(long,long,float*,long,float[]);
void snqqrflong,long,float*,long,LOGICAL32,long[],long,float[],float[],float[]);
void snqupd(long,long,float[],long,float[],float[],float[],LOGICAL32*);
float snrm2(long,float[],long);
void spfit(long,float[],float[],float[],long,LOGICAL32,LOGICAL32,LOGICAL32,float[],long*,float*,float*);
void splot(float,float,float[],long,float[],float[],STRING);
void splota(long);
void splote(long,float[],STRING);
void splotf(long,float[],float[],float[]);
void splotn(float,long,float[]);
void splott(long,float[]);
void splotr(float[],long,long,long);
void splot0(void);
void splot1(void);
void splot2(float,float,float,float);
void splot4(float,float,char*,char*);
void splot5(float,float,float,float);
void splot6(float,float,float,float,float);
void splot7(long*,long[],float[]);
void splot8(float,float,float,float,float,float,long,float);
void splot9(void);
void splotl(long,float[],float[]);

```

```

void    splots(float[],long);
void    spolz(float[],long,float[],float*,long*);
void    spolz2(float[],float[]);
float   sppnml(float,float,float);
float   spquad(long,long,float[],float*,float,float);
void    sprpl(float,byte,byte[],long,float,float,LOGICAL32);
void    sprpl1(float[],float[],long,char*,char*,char*,long,long,byte[],long*);
void    sprpl2(float*,long,long,long[],long[],long[],byte[],char*,char*,char*,long,long,byte[],long*);
void    sprpl3(float,float,float,float,float*,float*,float*,float*,long*,long*,long*,long*,char*,char*,char*,long,long,
    byte[],long*);
void    sprpl4(float,float,float*,float*,long*,long*,byte[6],long*,long*);
void    sprpl5(float,float,long,byte[6],long,long,long,long,byte[]);
void    sprtsv(float*,long,long,long,CHAR_INT,long,long,long);
float   spsi(float);
void    spsik(float,float,long);
void    spsie(float*,long*);
void    spsib(void);
float   spval(long,long,float[],float*,float,long);
void    sq7rfh(long*,long[],long,long,long,long,float*,float[],long,float[]);
void    ss7cpr(float[],long[],long,long);
void    sv7prm(long,long[],float[]);
void    sv7swp(long,float[],float[]);
void    sqrbd(long*,float[],float[],long,float*,long,long,float*,long,long);
float   srane(float);
float   srang(void);
void    srangv(float*,long,long,float[],float[],LOGICAL32*,long*);
float   sranr(float);
float   sranu(void);
float   srcomp(float,float);
void    srcval(float,float,float*,long*);
void    srdval(float,float,float,float*,long*);
float   srexpr(float);
void    srft(float[],byte,long[],long,long*,float[]);
void    srft1(float[],byte,long,long*,float[]);
void    srfval(float,float,float,float*,long*);
void    srfvlx(float,float,float,float*);
void    srjval(float,float,float,float,float*,long*);
float   srlog(float);
float   srlog1(float);
float   srlog2(float);
void    srn2g(float[],float*,long[],long,long,long,long,long*,long*,long,float[],float[],float[],float[]);
void    sg7lit(float[],float[],long[],long,long,long,long,float[],float[],float[]);
void    sn2lrd(float*,long[],float[],long,long,long,long,long,float[],float[],float[]);
void    sc7vfn(long[],float[],long,long,long,long,long,float[]);
void    sf7hes(float[],float[],long*,long[],long,long,long,float[],float[]);
void    sn2cvp(long[],long,long,long,float[]);
void    sn2rdp(long[],long,long,float[]);
void    so7prd(long,long,long,float[],float[],float*,float*);
void    sl7nvr(long,float[],float[]);
void    sl7tsq(long,float[],float[]);
void    srn2gb(float[][2],float[],float*,long[],long,long,long,long*,long*,long,float[],float[],float[],float[]);
void    sg7itb(float[][2],float[],float[],long[],long,long,long,long,float[],float[],float[]);
void    sr7tvm(long,long,float[],float[],float*,float[]);
void    sf7dhb(float[][2],float[],float[],long*,long[],long,long,long,float[],float[]);
float   sh2rfg(float,float,float*,float*,float*);
void    sh2rfa(long,float[],float[],float,float,float);
void    sg7qsb(float[][2],float[],float[],float[],long[],long[],long[],long*,float[],long,long,long*,long,float*,
    float[],float[],float[],float[],float[],float[]);
void    sl7msb(float[][2],float[],float[],long,long[],long[],long[],long*,float[],long,long,long*,long,float[],float[]);

```

```

    float*,float[],float[],float[],float[],float[],float[],float[]);
void ss7bqn(float[][2],float[],float[],long[],long[],long[],long*,float[],long,long*,long,long*,float[],float[],
    float[],float[],float[],float[],float[]);
void sq7rsh(long,long,LOGICAL32,float[],float[],float[]);
void sv7vmp(long,float[],float[],float[],long);
void sv7ipr(long,long[],float[]);
void sv7shf(long,long,float[]);
void ss7ipr(long,long[],float[]);
void sd7mlp(long,float[],float[],float[],long);
void ss7dmp(long,float[],float[],float[],long);
void srnsg(float*,float[],float[],float*,long[][2],long[],long,long,long,long,long,long,long,float[],float[]);
void srnsgb(float*,float[],float[][2],float[],float*,long[][2],long[],long,long,long,long,long,long,long,long,
    float[],float[]);
void srot(long,float[],long,float[],long,float,float);
void srotg(float*,float*,float*,float*);
void srotm(long,float[],long,float[],long,float[]);
void srotmg(float*,float*,float*,float,float[]);
void ssbasd(long,long,float[],float,long,float[]);
void ssbasi(long,long,float[],float,float,long*,long*,float[]);
void sscal(long,float,float[],long);
void ssdif(long,long,float[],float[],long,float*);
void ssfind(float[],long,long,float,long*,long*);
void ssfit(float[],float[],float[],long,long,long,float[],float[],float*,long*,long,float*);
void ssfitc(byte[][5],float[],float[],float[],long,long,float[],float[],float*,long[],long[],float[]);
float ssi(float);
float ssin1(float);
float ssinhm(float);
float ssinpx(float);
float ssnpxx(float);
void ssort(float[],long,long);
void ssortp(float[],long,long,long[]);
void ssortq(float[],long,long,long[]);
void sspge(long,long[],long[],float[],float[],float[]);
float ssquad(long,long,float[],float[],float,float);
void sstat1(float[],long,float[],long[],long,float,float);
void sstat2(float[],long[],long,float,float);
void sstop(long,long,float[],float[],float*,long*,float[],float*);
void ssva(float*,long,long,long,long,float[],float[],long[],CHAR_INT,long,float[],float[]);
float sssval(long,long,float[],float[],float,long);
void sssvala(long,long,float[],long,float*,float,float[]);
void ssvdrs(float*,long,long,long,float*,long,long,float[],float[]);
void sswap(long,float[],long,float[],long);
void ssymql(float*,long,long,float[],float[],long*);
void stcst(float[],char*,char*,long[],long,long*,float[]);
void stgc0(float[][3],float*,LOGICAL32,float[]);
void stgc1(LOGICAL32,float[][3],float*,LOGICAL32,float[]);
void stgext(float[],float[],float[],float[][2],long[],long[][4],long,long,float[],long,long,float*,LOGICAL32,float[]);
void stgqs(float[],long[],float[],float[],float[][3]);
void stgfi(float[],float[],float[],float[][2],long[],long,long[][4-(1)+1],long,long,float[],float*,LOGICAL32,
    float[],long*,float[]);
void stgfnd(float[],float[],long[],long,float[],long*,long[],float[][3],long*);
void stggrd(float[],float[],long,long[],float[],long[],long,long[][4],long,long*,long[]);
void stgupd(long,long,long[],long[],long);
float stgang(float,float,float);
void stgadaj(long,long,float[],float[],long,long[],long,long[][4],long,long,LOGICAL32*);
void stgpd(float[],float[],float[],float[][2],long,long[],long,long[]);
void stgmor(float[],float[],float[],long,long[],float[][21]);
void stgls(float[][21],long,long,LOGICAL32*,long,float*,float*);
void stgprg(float[],float[],long,long[],long[][4],long,long);

```



```

void stgrec(float[],float[],float[],float[][2],long,long[],long,long[][4],long,float[],long,long,float,float*,long,
long,long,LOGICAL32,float*);
void stgset(long,long,long,long,long,long,long,long[],long);
void stgget(long,long[],long[]);
void stgput(long,long[],long[],long);
void stgsiz(long,long*);
void strc2c(float[],long,float[],float[],float[]);
void susetn(long,long,long);
void sugetn(long*,long*,long*,long*,long*);
void suset(float,long,float[]);
void supro(float[],float[],float[]);
void suquo(float[],float[],float[]);
void susum(float[],float[],float[]);
void sudif(float[],float[],float[]);
void susum1(float,float[],float[]);
void sudif1(float,float[],float[]);
void supro1(float,float[],float[]);
void suquo1(float,float[],float[]);
void susqrt(float[],float[]);
void suexp(float[],float[]);
void sulog(float[],float[]);
void supwri(long,float[],float[]);
void susin(float[],float[]);
void sucos(float[],float[]);
void susinh(float[],float[]);
void sucosh(float[],float[]);
void suatan(float[],float[]);
void suatn2(float[],float[],float[]);
void suasin(float[],float[]);
void suacos(float[],float[]);
void suacs(LOGICAL32,float[],float[]);
void sutan(float[],float[]);
void sutanh(float[],float[]);
void surev(float*,float*,long,float*,long[],float*);
void svecp(float[],long,char*);
void svecpr(float[],long,char*,long,long,long);
void swatan(long,float[],float[]);
void swasin(long,float[],float[]);
void swacos(long,float[],float[]);
void swacsi(long,float[],float[],LOGICAL32);
void swatn2(long,float[],float[],float[]);
void swsum(long,float[],float[],float[]);
void swdif(long,float[],float[],float[]);
void swsqrt(long,float[],float[]);
void swexp(long,float[],float[]);
void swsin(long,float[],float[]);
void swcos(long,float[],float[]);
void swtan(long,float[],float[]);
void swsinh(long,float[],float[]);
void swcosh(long,float[],float[]);
void swtanh(long,float[],float[]);
void swset(long,float,float,float[]);
void swsum1(long,float,float[],float[]);
void swdif1(long,float,float[],float[]);
void swpro1(long,float,float[],float[]);
void swquo1(long,float,float[],float[]);
void swlog(long,float[],float[]);
void swpwri(long,long,float[],float[]);
void swchn(long,float[],float[]);

```

```

void swrchn(long,float[],float[]);
void swpro(long,float[],float[],float[]);
void swquo(long,float[],float[],float[]);
void spasc1(long,float[]);
float sxparg(long);
void sxrk8(float[],float[],float[],long[],float[],float[]);
void sxrk8a(float[],float[],float[],long[],float[],float[]);
void sxrk8i(float,float[],long[],float[]);
void sxrk8n(long[],float[],float[],float[],float[],float[],float[]);
float sxrk8x(float,float,float,float);
void sxrk8f(float*,float[],float[],long[]);
void sxrk8g(float[],float[],float[],long[]);
void sxrk8o(float[],float[],long[],float[]);
void szero(float*,float*,float*,float*,long*,float);
void umess(CHAR_INT,long[],long[]);
void xerbla(char*,long);
void zcoef(long,double[][2],double[][2]);
void zdif(double[],double[],double[]);
void zgam(double[],double[],double*,long);
void zpolz(double[][2],long,double[][2],double*,long*);
void zpro(double[],double[],double[]);
void zquo(double[],double[],double[]);
void zsqrtr(double[],double[]);
void zsum(double[],double[],double[]);
void zwofz(double[],double[],long*);

```