

```

#define gen_max 1000
#define D 5
#define NP 20
#define F 0.5
#define CR 0.8
#define inibound_l 27.0
#define inibound_h 45.0

/*----Constant for rnd_uni()-----*/
#define IM1 2147483563
#define IM2 2147483399
#define AM (1.0/IM1)
#define IMM1 (IM1-1)
#define IA1 40014
#define IA2 40692
#define IQ1 53668
#define IQ2 52774
#define IR1 12211
#define IR2 3791
#define NTAB 32
#define NDIV (1+IMM1/NTAB)
#define EPS1 1.2e-7
#define RNMIX (1.0-EPS1)

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

double evaluate(double [],long *);
double evaluate(double tmp[],long *nfe)
{
    double cost; (*nfe)++; /* tmp[0]=x1, tmp[1]=x2 tmp[2]=x3 */

    cost=(-5.357854*tmp[0]*tmp[0]-0.835689*(tmp[3])*tmp[2]-37.29329*(tmp[3])+40792.141);
    return(cost);

} /***** end of evaluate() *****/

float rnd_uni(long *);
float rnd_uni(long *idum)
{
    long j; long k;
    static long idum2=123456789;
    static long iy=0;static long iv[NTAB]; float temp;
    if(*idum<=0)
    {
        if(-(*idum)<1) *idum=1; else *idum=-(*idum); idum2=(*idum);
        for(j=NTAB+7;j>=0;j--)
        {
            k=(*idum)/IQ1;
            *idum=IA1*( *idum-k*IQ1)-k*IR1;
            if(*idum<0) *idum+=IM1;
            if(j<NTAB) iv[j]=*idum;
        }
        iy=iv[0];
    }
    k=(*idum)/IQ1;
    *idum=IA1*( *idum-k*IQ1)-k*IR1;
    if(*idum<0) *idum+=IM1;
    k=idum2/IQ2;
    idum2=IA2*( idum2-k*IQ2)-k*IR2;
    if(idum2<0) idum2+=IM2;
    j=iy/NDIV; iy=iv[j]-idum2; iv[j]=*idum;
    if(iy<1) iy+=IMM1;
    if((temp=AM*iy)>RNMIX) return RNMIX;
    else return temp;
}

void main()

{
    int i,j,k,a,b,c,good,count=0,seed; long nfe=0;
    double x1[50][10],x2[50][10],cost[50],trial[10],cost_trial,pen;
    double lhs2,lhs3,lhs4,costmin,lhs1,costmax;
    double a1=85.334407,a2=0.0056858,a3=0.0006262,a4=0.0022053,a5=80.51249,a6=0.0071317;

```

```

double a7=0.0029955,a8=0.0021813,a9=9.300961,a10=0.0047026,a11=0.0012547,a12=0.0019085;
clock_t start, end;
printf("\nseed=");
scanf("%d",&seed);
long rnd_uni_init= -(long)seed; start = clock();

for (i=0;i<NP;i++)
{
    for (j=0;j<D;j++)
    {
        if(j==3)
            x1[i][j]=78.0 + rnd_uni(&rnd_uni_init)*(102.0-78.0);
        else
            x1[i][j]=inibound_l + rnd_uni(&rnd_uni_init)*(inibound_h-inibound_l);
        }
    pen=0.0;

    lhs1=a1+a2*(x1[i][4])*x1[i][2]+a3*x1[i][1]*(x1[i][3])-a4*x1[i][0]*x1[i][2];
    if(lhs1>92.0)
    {
        pen=lhs1*10.0;
        cost[i]=evaluate(x1[i], &nfe);
        cost[i]=cost[i]-pen;
        continue;
    }

    lhs2=a5+a6*(x1[i][4])*x1[i][2]+a7*(x1[i][4])*(x1[i][3])+a8*x1[i][0]*x1[i][0]-90.0;
    if(lhs2>20.0)
    {
        pen=lhs2*10.0;
        cost[i]=evaluate(x1[i], &nfe);
        cost[i]=cost[i]-pen;
        continue;
    }

    lhs3=a9+a10*x1[i][0]*x1[i][2]+a11*x1[i][0]*(x1[i][3])+a12*x1[i][0]*x1[i][1]-20.0;
    if(lhs3>5.0)
    {
        pen=lhs3*10.0;
        cost[i]=evaluate(x1[i], &nfe);
        cost[i]=cost[i]-pen;
        continue;
    }

    if(lhs1<=92.0 && lhs2<=20.0 && lhs3<=5.0)
        cost[i]=evaluate(x1[i], &nfe);
}

costmax=cost[0];
for(i=1;i<NP;i++)
{ if(costmax<cost[i])
costmax=cost[i];
}

while (count<gen_max)
{
    for (i=0;i<NP;i++)
    {
        do a=int ((rnd_uni(&rnd_uni_init))*NP); while (a==i);

        do b=int (rnd_uni(&rnd_uni_init)*NP); while (b==i || b==a);

        do c=int (rnd_uni(&rnd_uni_init)*NP); while (c==i || c==a || c==b);

        j=int (rnd_uni(&rnd_uni_init)*D);

        for (k=1;k<=D;k++)
        {
            if(rnd_uni(&rnd_uni_init)<CR || k==D)
            {
                trial[j]=x1[c][j]+F*(x1[a][j]-x1[b][j]);
            }
            else trial[j]=x1[i][j];
            if(j==3)
            { if(trial[3]<78.0) trial[3]=(78.0);
              if(trial[3]>102.0) trial[3]=(102.0);
            }
        }
    }
}

```

```

        }
        else
        {
            if(trial[j]<27.0)    trial[j]=(27.0);
            if(trial[j]>45.0)    trial[j]=(45.0);
        }
        j=(j+1)%D;
    }

pen=0.0;

lhs1=a1+a2*trial[2]*(trial[4])+a3*trial[1]*(trial[3])-a4*trial[0]*trial[2];
if(lhs1>92.0)
{
    pen=lhs1*10.0;
    cost_trial=evaluate(trial, &nfe);
    cost_trial=cost_trial-pen;
    if(cost_trial>=cost[i])
    {
        for (j=0;j<D;j++)
            x2[i][j]=trial[j];
        cost[i]=cost_trial;
        if(cost_trial>costmax)
        {
            costmax=cost_trial;
            /* imin=i;
            assignd(best,trial); */
        }
    }
    else for (j=0;j<D;j++)
        x2[i][j]=x1[i][j];
    continue;
}

lhs2=a5+a6*trial[2]*(trial[4])+a7*(trial[4])*(trial[3])+a8*trial[0]*trial[0]-90.0;
if(lhs2>20.0)

{
    pen=lhs2*10.0;
    cost_trial=evaluate(trial, &nfe);
    cost_trial=cost_trial-pen;
    if(cost_trial>=cost[i])
    {
        for (j=0;j<D;j++)
            x2[i][j]=trial[j];
        cost[i]=cost_trial;
        if(cost_trial>costmax)
        {
            costmax=cost_trial;
            /* imin=i;
            assignd(best,trial); */
        }
    }
    else for (j=0;j<D;j++)
        x2[i][j]=x1[i][j];
    continue;
}

lhs3=a9+a10*trial[2]*trial[0]+a11*trial[0]*(trial[3])+a12*trial[0]*trial[1]-20.0;
if(lhs3>5.0)

{
    pen=lhs3*10.0;
    cost_trial=evaluate(trial, &nfe);
    cost_trial=cost_trial-pen;
    if(cost_trial>=cost[i])
    {
        for (j=0;j<D;j++)
            x2[i][j]=trial[j];
        cost[i]=cost_trial;
        if(cost_trial>costmax)
        {
            costmax=cost_trial;
            /* imin=i;
            assignd(best,trial); */
        }
    }
    else for (j=0;j<D;j++)

```

```

        x2[i][j]=x1[i][j];
    continue;
}

if(lhs1<=92.0 && lhs2<=20.0 && lhs3<=5.0)
{
    cost_trial=evaluate(trial, &nfe);
    if(cost_trial>=cost[i])
    {
        for (j=0;j<D;j++)
            x2[i][j]=trial[j];
        cost[i]=cost_trial;
        if(cost_trial>costmax)
        {
            costmax=cost_trial;
            /* imin=i;
            assignd(best,trial); */
        }
    }
    else for (j=0;j<D;j++)
        x2[i][j]=x1[i][j];
    continue;
}
} /***** end of for loop *****/

for (i=0;i<NP;i++)
{
    for (j=0;j<D;j++)
        x1[i][j]=x2[i][j];
}

costmax=cost[0];
for(i=1;i<NP;i++)
{ if(costmax<cost[i])
costmax=cost[i];
}
costmin=cost[0];
for(i=1;i<NP;i++)
{ if(costmin>cost[i])
costmin=cost[i];
}

if((costmax-costmin)<0.00001)
    break;
count++;

} /***** end of while loop *****/

    end = clock();
for(i=0;i<NP;i++)
{
    for(j=0;j<D;j++)
        printf("u[%d]=%lf  ",j, (x1[i][j]));

    printf("cost[%d]=%lf  ",i,cost[i]);
}
printf("NFE=%ld\n",nfe);
printf("The time was: %f\n", (end - start) / CLK_TCK);
printf("costmax is=%lf",costmax);
} /***** end of main() *****/

```