

10.Bölüm

Başlangıç Örnekler 1

```
*****
faktoryel hesaplayan program
*****

#include<stdio.h>
/* 1 den 100 e kadar olan sayilarin faktoriyeli */
main()
{ int a,i;
double f;
f=1;
clrscr();
for (a=1;a<101;a++)
{ f=f*a;
printf(" %d faktoriyel %f \n",a,f);
i++;
if (i>10)
{i=1;
getch(); }
}
getch();
}

*****
*
Asal sayıları bulan program
*****
*
/*      BIRDEN 2000 E KADAR ASAL SAYILARI BULAN PROGRAM */
#include <stdio.h>
#include <math.h>
float tam(sayi)
float sayi;
{
int i;
i=sayi;
return i+.0;
}
int asal(sayi)
int sayi;
{
```

```

int i,sonuc;
sonuc=1;
for (i=2;i<sayi/2+1;i++)
{
if ((sayi+.0)/i==tam((sayi+.0)/i))
{
sonuc=0;
i=sayi;
}
}
if (sayi==-1 ||sayi==0||sayi==1) sonuc=0;
return sonuc;
}
main()
{
int j;
clrscr();
for(j=2;j<2000;j++)
if (asal(j)==1) printf("%4d",j);
getch();
}

```

e değerini hesaplayan program

```

#include <stdio.h>
#include<math.h>
double fakt(sayi)
int sayi;
{
int i;
double fak=1.0;
for (i=1;i<sayi+1;i++)
fak=fak*i;
return fak;
}
double expu(x)
float x;
{
int i;
double e=0.0;
for (i=0;i<41;i++)

```

```

e=e+pow(x,i)/fakt(i);
return e;
}
main()
{
float x;
clrscr();
printf("Exp i alınacak sayiyi giriniz..:");
scanf("%f",&x);
printf("Yaklaşık exp(%f)=%.20f\n",x,expu(x));
printf("program fonksiyonu exp(%f)=%.20f",x,exp(x));
getch();
}

```

Bazı betimsel istatistikleri hesaplayan program

```

#include<stdio.h>
#include<math.h>
# define boyut 1000
main()
{
int t,a,n,i,j;
int dizi[boyut], max,min,gen;
double var,ort,ss,med;
int vr;
clrscr();
printf("dizi kac elemanli = ");
scanf("%d",&n);
printf("elemanlari girin \n ");
for (i=1;i<n+1;i++)
{ printf(" %d",i);printf(".eleman=");
scanf("%d",&dizi[i]);}
/* siralama */
for(i=1;i<n;i++)
for(j=i+1;j<n+1;j++)
if (dizi[i]>dizi[j])
{ a=dizi[i];
dizi[i]=dizi[j];
dizi[j]=a;
}
printf("SIRALI HALi.. ");
for(i=1;i<n+1;i++)

```

```

printf("%d ",dizi[i]);
printf("\n");
/* max min */
max=dizi[n];min=dizi[1];gen=max-min;
printf("maximum= %d \n",max);
printf("minimum= %d \n",min);
printf("orneklem genisligi= %d \n",gen);
/* ortalama s.sapma varyans*/
t=vr=0;ort=var=ss=0;
for(i=1;i<n+1;i++)
    t=t+dizi[i];
    ort=t/n;
for(i=1;i<n+1;i++)
vr=vr+pow((dizi[i]-ort),2);
var=vr/(n-1);
ss=sqrt(var);
printf("ortalama %f \n",ort);
printf("varyans %f \n",var);
printf("s.sapma %f \n",ss);
printf("ortanca ");
if (pow(-1,n)<0) med=dizi[(n+1)/2];
else med=(dizi[n/2]+dizi[n/2+1])/2;
printf("%f \n",med);
getch();
}

*****
program adi : kokbul.c
Çeşitli fonksiyonların köklerini değişik yöntemlerle bulan program
*****

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

int e=2.71182818;
char *dizi[14];
void aktar(void)
{
int sayac;
int graphdriver=DETECT,graphmode;

```

```

initgraph(&graphdriver,&graphmode,"");
setbkcolor(1);
setcolor(14);
settextstyle (GOTHIC_FONT,HORIZ_DIR,1);
setcolor(11);
setlinestyle(0,0,3);
settextstyle(DEFAULT_FONT,HORIZ_DIR,2);
outtextxy(250,15,"SORULAR");
gotoxy(1,1);printf("Selcuk Cobanoglu");
gotoxy(4,2);printf("95052732");
line(0,50,700,50);
settextstyle(SMALL_FONT,HORIZ_DIR,7);

dizi[1]=" X-SIN(X)-1=0 ";
dizi[2]=" X-e^(-X)=0 ";
dizi[3]=" X^3+10X-11.6=0 ";
dizi[4]=" X*e^X-1=0 ";
dizi[5]=" X^3-100=0 ";
dizi[6]=" e^X-3*X=0 ";
dizi[7]=" X*TAN(X)-0.5=0 ";
dizi[8]=" COS(X)=3*X ";
dizi[9]=" X^3-3X+1=1 ; Xo=1.5 ";
dizi[10]=" e^X-LN(X)=20 ";
dizi[11]=" e^X=(2*X)+21 ; Xo=3 ";
dizi[12]=" LN(X)=1+1/(X^2) ";
{
settextstyle(SMALL_FONT,HORIZ_DIR,1);
setcolor(1);
for (sayac=1;sayac<13;sayac++)
{
gotoxy(30,4+sayac);printf("%d",sayac);printf(dizi[sayac]);
printf("\n");
}
}
}
/*****/
double fonk(double x,int fno)
{
double f1;
switch(fno)
{
case 1:{f1= x-sin(x)-1;break;}
case 2:{f1= x-pow(e,-1*x);break;}
case 3:{f1= pow(x,3)+10*x-11.6;break;}

```

```

case 4:{f1= x*pow(e,x)-1;break;}
case 5:{f1= pow(x,3)-100;break;}
case 6:{f1= pow(e,x)-3*x;break;}
case 7:{f1= x*tan(x)-0.5;break;}
case 8:{f1= cos(x)-3*x;break;}
case 9:{f1= pow(x,3)-3*x;break;}
case 10:{f1= pow(e,x)-log(x)-20;break;}
case 11:{f1= pow(e,x)-2*x-21;break;}
case 12:{f1= log(x)-1-1/pow(x,2);break;}
}
return f1;
}
/*****/
double turev(double x,int fno)
{
double t1;
switch(fno)
{
case 1:{t1= 1-cos(x);break;}
case 2:{t1= 1+pow(e,-1*x);break;}
case 3:{t1= 3*pow(x,2)+10;break;}
case 4:{t1= pow(e,x)+x*pow(e,x);break;}
case 5:{t1= 3*pow(x,2);break;}
case 6:{t1= pow(e,x)-3;break;}
case 7:{t1= tan(x)+x*(1+pow(tan(x),2));break;}
case 8:{t1= -1*sin(x)-3;break;}
case 9:{t1= 3*pow(x,2)-3;break;}
case 10:{t1= pow(e,x)-1/x;break;}
case 11:{t1= pow(e,x)-2;break;}
case 12:{t1= 1/x+2/pow(x,3);break;}
}
return t1;
}
/*****/
double ifonk(double x,int fno)
{
double f1;
switch(fno)
{
case 1:{f1= sin(x)+1;break;}
case 2:{f1= pow(e,-1*x);break;}
case 3:{f1= (11.6-pow(x,3))/10;break;}
case 4:{f1= pow(e,-1*x);break;}
case 5:{f1= 10/pow(x,0.5);break;}

```

```

case 6:{f1= log(3*x);break;}
case 7:{f1= atan(1/(2*x));break;}
case 8:{f1= cos(x)/3;break;}
case 9:{f1= pow(x,3)-3*x;break;}
case 10:{f1= log(log(x)+20);break;}
case 11:{f1= log(2*x+21);break;}
case 12:{f1= pow(e,1+1/(x*x));break;}
}
return f1;
}
/*****
void cevap(no,yontem)
int no,yontem;
{
int sayac;
float xx,x,x1,x2,eps;
printf("Xo 'i Giriniz..");
scanf("%f",&xx);
printf("Epsilon'u Giriniz");
scanf("%f",&eps);
printf("\n");
if (yontem==1 && no!=10 && no!=12)
{
/****YARILAMA YONTEMI*****/
sayac=1;
x1=-1*5;
x2=5;
x=(x1+x2)/2;
printf("Xo=%f\n",x);
while (-1*eps>=fonk(x,no) || fonk(x,no)>=eps)
{
if (fonk(x1,no)*fonk(x,no)<0) x2=x; else x1=x;
x=(x1+x2)/2;
printf("X%d=%f\n",sayac,x);
sayac++;
}
printf("\nE%d=%.6f\n",sayac-1,fonk(x,no));
}
if (yontem ==2 && no!=10 && no!=6 && no !=12)
{
/****REGULA FALSI YONTEMI*****/
sayac=1;
x1=-1*10;
x2=5;

```

```

x=(x1+x2)/2;
printf("Xo=%f\n",x);
while (-1*eps>=fonk(x,no) || fonk(x,no)>=eps)
{
    if (fonk(x1,no)*fonk(x,no)<0) x2=x; else x1=x;
    x=(x1*funk(x2,no)-x2*funk(x1,no))/(fonk(x2,no)-fonk(x1,no));
printf("X%d=%f\n",sayac,x);
sayac++;
}
printf("\nE%d=%.6f\n",sayac-1,funk(x,no));
}
if (yontem==3)
{
/** NEWTON RAPSON YONTEMI***/
sayac=1;
x=xx;
printf("Xo= %f\n",x);
while (-1*eps>=fonk(x,no) || fonk(x,no)>=eps)
{
    x=x-(fonk(x,no)/turev(x,no));
printf("X%d= %.6f\n",sayac,x);
    x1=x;
}
printf("\nE%d=%.6f\n",sayac-1,ifonk(x,no));
}
if (yontem==4)
{
/**BASIT ITERASYON YONTEMI*****/
sayac=1;
x=xx;
printf("Xo= %f\n",x);
while (-1*eps>=x-ifonk(x,no) || x-ifonk(x,no)>=eps)
{
x=fonk(x,no);
printf("X%d= %.6f\n",sayac,x);
}
printf("\nE%d=%.6f\n",sayac-1,x-ifonk(x,no));
}
}
/*****/
main()
{
int s,y;
aktar();

```

```

gotoxy(45,40);printf("Secenek Gir..");
scanf("%d",&s);
gotoxy(50,50);printf(" \n YONTEMLER \n");
gotoxy(50,51);printf("1 => YARILAMA\n");
gotoxy(50,52);printf("2 => REGULA FALSI\n");
gotoxy(50,53);printf("3 => NEWTON RAPSON\n");
gotoxy(50,54);printf("4 => BASIT ITERASYON\n\n");
gotoxy(50,55);printf("Secenek Giriniz..");
scanf("%d",&y);
if (s>0 && s<14) cevap(s,y);
getch();
}

```

matrislerle ilgili işlem yapan program

```
#include<stdio.h>
```

```
/* matrislerin toplami max min toplami max-min satir top min olan sutun*/
```

```
main()
```

```
{
```

```
int n;
```

```
int ma[100][100],mb[100][100],mc[100][100];
```

```
int i,j,k,s;
```

```
int maxa,maxb,mina,minb,satir[100],sutun[100];
```

```
int iza[100][100],izb[100][100],a[100],b[100];
```

```
clrscr();
```

```
printf("n= ");
```

```
scanf("%d",&n);
```

```
for (i=1;i<n+1;i++)
```

```
for (j=1;j<n+1;j++)
```

```
ma[i][j]=0;
```

```
mb[i][j]=0;
```

```
mc[i][j]=0;
```

```
printf("A matrisi (nxn) \n");
```

```
for(i=1;i<n+1;i++)
```

```
for(j=1;j<n+1;j++)
```

```
scanf("%d",&ma[i][j]);
```

```
printf("B matrisi (nxn) \n");
```

```
for(i=1;i<n+1;i++)
```

```
for(j=1;j<n+1;j++)
```

```
scanf("%d",&mb[i][j]);
```

```
for(i=1;i<n+1;i++)
```

```

for(j=1;j<n+1;j++)

    mc[i][j]=ma[i][j]+mb[i][j];
for(i=1;i<n+1;i++)
for(j=1;j<n+1;j++)
    printf("%d ",mc[i][j]);
    printf(" \n ");
/* max - min bulma */
maxa=ma[1][1];mina=ma[1][1];
maxb=mb[1][1];minb=mb[1][1];
for(i=1;i<n+1;i++)
for(j=1;j<n+1;j++)
{if (maxa<ma[i][j]) maxa=ma[i][j];
if (maxb<mb[i][j]) maxb=mb[i][j];
if (mina>ma[i][j]) minb=ma[i][j];
if (minb>mb[i][j]) minb=mb[i][j];}
printf("a matrisinin en buyuk degeri= %d \n",maxa);
printf("b matrisinin en buyuk degeri= %d \n",maxb);
printf("a matrisinin en kucuk degeri= %d \n",mina);
printf("b matrisinin en kucuk degeri= %d \n",minb);
/* satir ve sutunda max min */
for(i=1;i<100;i++)
{satir[i]=0;sutun[i]=0; }
for(i=1;i<n+1;i++)
for(j=1;j<n+1;j++)
    { satir[i]=satir[i]+ma[i][j];
    sutun[i]=sutun[i]+mb[j][i];}
    maxa=0;minb=0;
    maxa=satir[1];minb=sutun[1];

    for (i=1;i<n+1;i++)
    { if(maxa<satir[i]) maxa=satir[i];
    if(minb>sutun[i]) minb=sutun[i];}
    printf("toplami max olan a matrisi satiri= %d \n",maxa);
    printf("toplami min olan b matrisi sutunu= %d \n",minb);
/* a ve b matrisinin kosegenleri */
s=0;k=0;
for(i=1;i<n+1;i++)
{a[i]=ma[i][i];
b[i]=mb[i][i];}

printf("a matrisi kosegenleri \n");
for(i=1;i<n+1;i++)
{ k=k+a[i];

```

```
printf("%d \n",a[i]);}
printf("kosegen toplami= %d \n",k);
printf("b matrisi kosegenleri \n");
for (i=1;i<n+1;i++)
{ s=s+b[i];
printf("%d \n",b[i]); }
printf("kosegen toplami= %d \n",s);
/* matrisin transpozu*/
for(i=1;i<n+1;i++)
for(j=1;j<n+1;j++)
{ iza[j][i]=ma[i][j];
izb[j][i]=mb[i][j]; }
printf("a nin izi \n");
for(i=1;i<n+1;i++)
for(j=1;j<n+1;j++)
printf(" %d ",iza[i][j]);
printf("\n");
printf("b nin izi: \n");
for(i=1;i<n+1;i++)
for(j=1;j<n+1;j++)
printf(" %d ",izb[i][j]);

getch();

}
```