

## **BÖLÜM 5**

### **POINTER'LER (İŞARETÇİLER)**

Bir programın çalışması sırasında bütün bilgiler ve program kodları bellek alanlarına yerleşir.

Bu bilgilere ulaşmak için değişken adı ya da fonksiyon isimleri kullanılır.

Değişken bildirimleriyle; uygun bir bellek alanının, bir isme ayrılıp kullanılması sağlanır.

Değişkenler değer (sayı, karakter, vs...) tutar. Pointer'lar ise adres tutan değişkenlerdir.

Pointer kullanarak (isimler yerine) bellek alanlarındaki bilgi işlenebilir.

Pointerler; hız ve bellek tasarrufu, dizi ve stringler üzerinde geliştirilmiş işlemler, boş bellek kullanımı, dinamik bellek kullanımı vb. gibi avantajlar sağlar.

## **Pointer değişkenler \* karakteriyle tanımlanır.**

Pointerler başka değişkenlere ait adres değerlerini içerir ve bu nedenle pointer değişken hangi tipte tanımlanırsa tanımlansın içeriği daima işaretsiz tamsayıdır. Ancak farklı tipteki değişken alanlarını gösterebilir.

Pointer tanımlamada kullanılan tip bildirisi, pointer değişkenin tipini değil pointerin göstereceği alanın tipini gösterir ve pointer tipi göstereceği alanın tipi ile uyumlu olmalıdır.

Örneğin

```
int *p;  
float *q  
char *x
```

## Örnek Değişkenlerde pointer atamaları

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

```
int main(void) {
```

```
int a,*p;
```

```
float x,*q;
```

```
char *k,y;
```

```
    p=&a; q=&x;
```

```
    *p=10;
```

```
    x=10.0/3; // ya da *q=10.0/3;
```

```
    /*(p+1)=3;
```

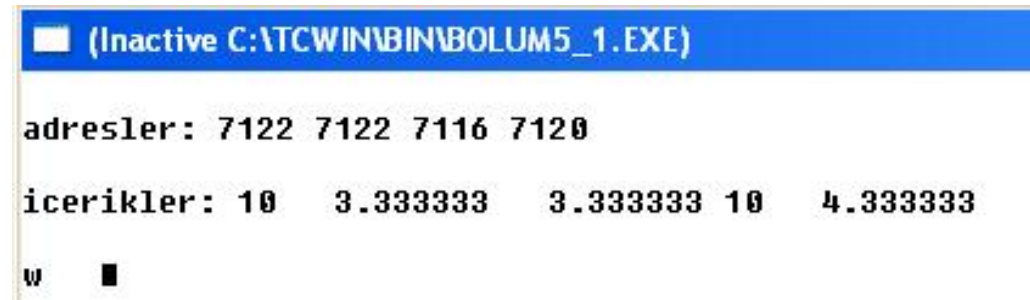
```
printf("\nadresler: %u %u %u %u\n",p,&a,q,&p);
```

```
printf("\nicerikler: %d %f %f %d %f\n",*p,*q,x,a,*q+1);
```

```
k=&y;
```

```
y='w';
```

```
printf("\n%c %c \n",y,*k);}
```



```
(Inactive C:\TCWIN\BIN\BOLUM5_1.EXE)
adresler: 7122 7122 7116 7120
icerikler: 10 3.333333 3.333333 10 4.333333
w ■
```

## Örnek Fonksiyonlara pointerle değer gönderme

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

```
#include<conio.h>
```

```
void deneme (int*,float*);
```

```
main() {
```

```
    int i=3;
```

```
    float j=10./3;
```

```
    printf("\n%d  %f",i,j);
```

```
    deneme(&i,&j);
```

```
    printf("\n%d  %f",i,j);
```

```
    getch();
```

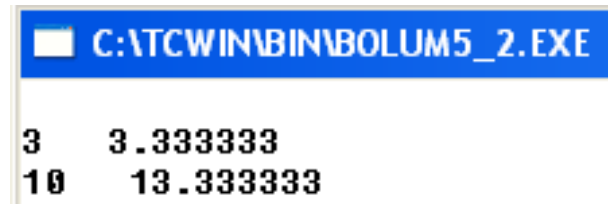
```
}
```

```
void deneme (int *p,float *q) {
```

```
*p=10;
```

```
*q=*p+*q;
```

```
}
```



```
C:\TCWIN\BIN\BOLUM5_2.EXE
3 3.333333
10 13.333333
```

## Örnek Fonksiyonlara pointerle değer gönderme

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

```
#include<conio.h>
```

```
int bolme ( int bolunen, int bolen, int *kalan )
```

```
{
```

```
    *kalan = bolunen % bolen;
```

```
    return bolunen / bolen;
```

```
}
```

```
int main( void ) {
```

```
    int bolunen, bolen;
```

```
    int bolum, kalan;
```

```
    bolunen = 13; bolen = 4;
```

```
    bolum = bolme( bolunen, bolen, &kalan );
```

```
    printf( "Bölüm: %d Kalan: %d\n", bolum, kalan );
```

```
    return 0;
```

```
}
```

## Örnek Fonksiyonlara pointerle değer gönderme

```
#include<stdio.h>
void degistir( int *x, int *y )
    {
    int temp;
    temp = *x; *x = *y; *y = temp;
    }

int main( void ) {
int a, b; a = 12; b = 27;
printf( "a: %d b: %d\n", a, b );
degistir(&a, &b);
printf( "a: %d b: %d\n", a, b ); return 0;
    }
```

## Örnek Dizilerde ve stringlerde Pointerler

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

```
void main(void) {
```

```
    int a[]={3,-5,9,20};
```

```
    int *p;
```

```
    char *q,s[]="Firat";
```

```
    p=a;                //ya da p=&a[0];
```

```
    q=s;                //ya da q=&s[0];
```

```
    printf("\n%d  %d  %d %s %c %s\n",*p,*(p+1),*p+3,q,*(q+1),q+2);  
}
```



```
(Inactive C:\TCWIN\BIN\BOLUM5_3.EXE)  
3 -5 6 Firat i rat
```

**Örnek** Klavyeden büyük harfle girilen stringi küçük harfe çeviren program.

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

```
main() { char s[80],*p;
        p=&s[0];
        printf("Buyuk harfleri kullanarak bir kelime giriniz=");
        scanf("%s",s);
        while(*p) printf("%c",tolower(*p++));
        getch(); }
```

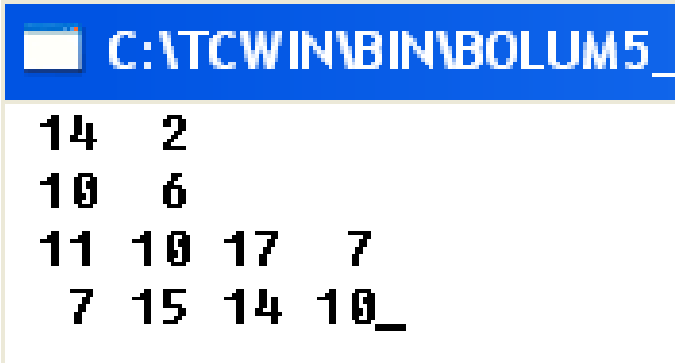


```
C:\TCWIN\BIN\BOLUM5_4.EXE
Buyuk harfleri kullanarak bir kelime giriniz=TEKNOLOJİ
teknoloji
```



**Örnek** Aşağıdaki programın çıktısını inceleyiniz.

```
# include <stdio.h>
# include <conio.h>
main(){
    int a[5]={2,6,10,14,18},*p;
    int *q;
    p=&a[3]; q=a;
    printf ("%3d%3d\n",*p,*q);
    p--;q++;
    printf("%3d%3d\n",*p,*q);
    printf("%3d%3d%3d%3d\n",*p+1,*(q+1),*p+*q,++*q);
    printf("%3d%3d%3d%3d",*q,*p,(*p)++,*p++);
    getch();
}
```



```
C:\TCWIN\BIN\BOLUM5_
14  2
10  6
11 10 17  7
 7 15 14 10_
```

NOT: 3. çıktı satırındaki p ve q pointer artımlarına dikkat.

## Örnek Fonksiyonlara dizi ve string gönderme

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

void topla (char *p) {
*(p+2)='D'; }

main () {
char s[]="Ahmet";
//topla(s);
topla(&s[0]);
printf ("%s ",s);
}
```

```
#include <stdio.h>
#include <conio.h>
int topla(int *);

void main(void){
int a[2]={3,10},c;
c=topla(a); //c=topla(&a[0]);
printf ("%d",c); }

int topla (int *p) {
int s;
*p=5;
s=*p+*(p+1);
return(s); }
```

**Örnek** strcat() fonksiyonun görevini yapan bir alt fonksiyon yazın.

```
# include <stdio.h>
# include <conio.h>
strekle(char *p,char *q){
    while(*p) p++;
    while(*q) {*p=*q;
    p++;q++;}
    *p=NULL;}
main() {char s1[15]="firat"; char s2[15]="unv";
strekle(s1,s2);
puts(s1); getch();}
```



C:\TCWIN\BIN\BOLUM5\_7.EXE  
firatunv

## Dizi Pointerler

Pointerler dizi olarak da tanımlanabilirler ve pointer dizisinin her bir elemanı farklı bir değişken alanını (genellikle dizi ve stringlerin her bir satırını) point edebilir. Stringler üzerindeki kullanımı daha kolaydır.

### Örnek

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



```
(Inactive C:\TCWIN\BIN\BOLUM5_8.EXE)
4 6 10 0
Ali Veli C
```

```
int *p[3][2]={ {2,4},{6,8},{10,11} }; //pointer atamasinda warning verebilir.
```

```
char *liste[4]={"Ali","Veli","Can","Osman"};
```

```
void main(void) {
printf("%d %d %d %d\n",*(p[0]+1),*p[1],*p[2],*p[0][0]);
printf("%s %s %c",liste[0],liste[1],liste[2][0]);}
```

**Örnek** Bir önceki örnekle sonuçlarını kıyaslayınız

```
int *p[3],a[3][2]={ {2,4},{6,8},{10,11} };  
char *liste[4],isim[4][10]={"Ali","Veli","Can","Osman"};
```

```
void main(void) {  
    p[0]=a[0];  
    p[1]=a[1];  
    p[2]=&a[2][0];  
    printf("%d %d %d\n",*(p[0]+1),*p[1],*p[2]);  
  
    liste[0]=isim[0];  
    liste[1]=isim[1];  
    liste[2]=&isim[2][0];  
    printf("%s %s %c",liste[0],liste[1],liste[2][0]);  
}
```



```
(Inactive C:\TCWIN\BIN\BOLUM5_8.EXE)  
4 6 10 0  
Ali Veli C
```

## Örnek

```
# include <stdio.h>
# include <conio.h>
#include<string.h>
int i=0,j,k,n=10;
char *liste[10],isim[10][15],*yedeK;
void main(void) {
while(i<n){
printf("%d. string=",i+1);scanf("%s",isim[i]);
    if (strcmp(isim[i],"son")==0) break;
    liste[i++]=isim[i];
    }
for (j=0;j<i-1;j++)
    for (k=j+1;k<i;k++)
        if(strcmp(liste[j],liste[k])>0){
            yedeK=liste[k];
            liste[k]=liste[j];
            liste[j]=yedeK; }
printf("\nSiralanmis isimler\n");
for(j=0;j<i;j++) puts(liste[j]);
}
```

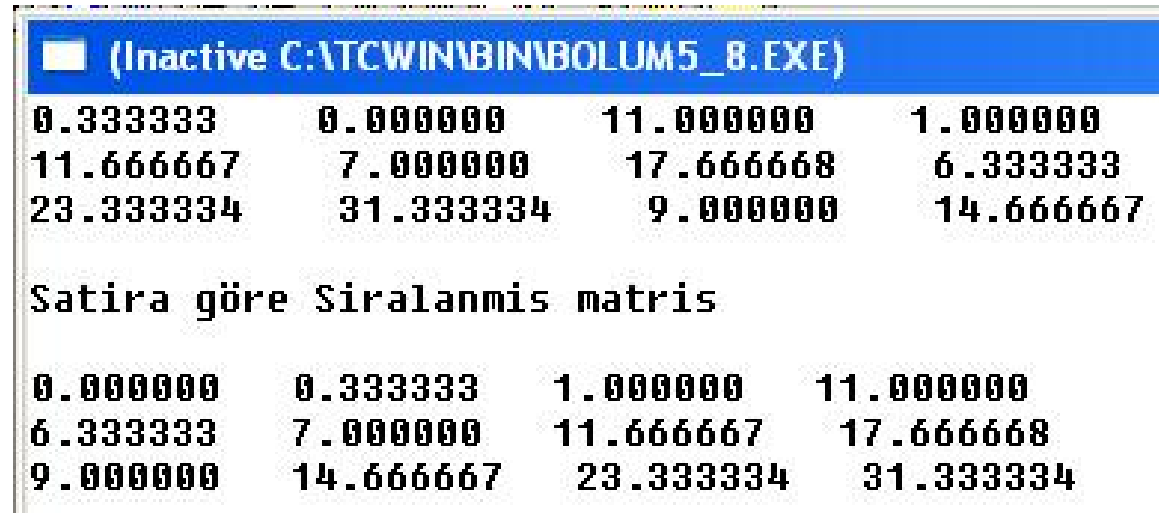
(Inactive C:\TCWIN\BIN\BOLUM5\_8.EXE)

```
1. string=Murat
2. string=Can
3. string=Ueli
4. string=Mehmet
5. string=son
```

```
Siralanmis isimler
Can
Mehmet
Murat
Ueli
```

## Örnek

```
# include <stdio.h>
#include<stdlib.h>
int i,j,k; float *p[3],x,d[3][4];
void main(void) {
    for (i=0;i<3;i++){
        for (j=0;j<4;j++){
            d[i][j]=(float)1/3*random(100);
            printf("%2.6f  ",d[i][j]); }
        printf("\n");
        p[i]=d[i];
    }
    for (i=0;i<3;i++)
        for (j=0;j<4;j++)
            for(k=j+1;k<4;k++)
                if(*(p[i]+j)>*(p[i]+k)){
                    x=*(p[i]+j);
                    *(p[i]+j)=*(p[i]+k);
                    *(p[i]+k)=x; }
    printf("\nSatira göre Sıralanmis matris\n");
    for (i=0;i<3;i++){
        printf("\n");
        for (j=0;j<4;j++) printf("%2.6f  ",d[i][j]); } }
```



```
(Inactive C:\TCWIN\BIN\BOLUM5_8.EXE)
0.333333  0.000000  11.000000  1.000000
11.666667  7.000000  17.666668  6.333333
23.333334  31.333334  9.000000  14.666667

Satira göre Sıralanmis matris

0.000000  0.333333  1.000000  11.000000
6.333333  7.000000  11.666667  17.666668
9.000000  14.666667  23.333334  31.333334
```

## Pointer Pointerleri

Bir pointer değişkeni, bir başka pointer değişkenini de point edebilir. Bu pointerler **\*\*** ile tanımlanır.

### Örnek

```
#include <stdio.h>
#include <conio.h>
main() { int d,*p,**q;
d=568;
```

```
p=&d; q=&p;          /*p=d; **q=*p;
```

```
printf ("\n%d  %d  %d",d,*p,**q); }
```



```
(Inactive C:\TCWIN\BIN\BOLUM512.EXE)
568 568 568
```



## Örnek Matrislerde pointer-pointeri

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

```
void main(void) {
```

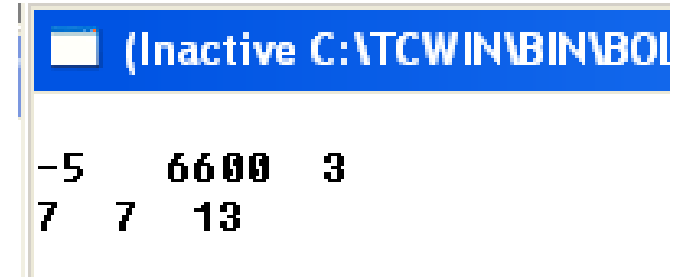
```
int a,d[2][3]={{3,-5,9},{2,7,6}},*p,**q;
```

```
p=d[0];q=&p;
```

```
printf ("\n%d  %d  %d",*(p+1),*q,**q);
```

```
printf ("\n%d  %d  %d",>(*q+4),*(p+4),**q+10);
```

```
}
```



```
(Inactive C:\TCWIN\BIN\BOI...
-5 6600 3
7 7 13
```

## Örnek

```
# include <stdio.h>
#include<stdlib.h>
#define m 2
#define n 3
void main(void) {
    int mat[m][n];
    int i,j;
        for (i=0;i<m;i++){
            for (j=0;j<n;j++){
                mat[i][j]=random(100);
                printf("%5d",mat[i][j]);}
    printf("\n"); }
    for (i=0;i<m;i++)
        for (j=0;j<n;j++)
            (*(mat+i)+j)=*(*(mat+i)+j)+10;
    printf("\n Yeni matris\n");
    for (i=0;i<m;i++){
        printf("\n");
        for (j=0;j<n;j++)
            printf("%5d",*(*(mat+i)+j));} }
```

 (Inactive C:\TCWIN\BIN\

```
1 0 33
3 35 21
```

Yeni matris

```
11 10 43
13 45 31
```