

CENG 111 ALGORİTMALAR VE PROGRAMLAMA

Prof. Dr. Tufan TURACI

tturaci@pau.edu.tr

- Pamukkale Üniversitesi
- Mühendislik Fakültesi
- Bilgisayar Mühendisliği Bölümü
- Hafta 14

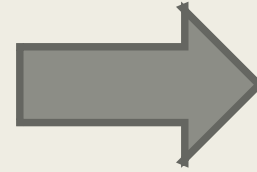
Dosyalar (Files)


Soru1: İçerisinde öğrenci ismi, numarası, vize ve final notu bilgileri bulunan 'ogrenci.txt' isimli bir dosya oluşturunuz. Daha sonra oluşturulan bu dosyadan tüm bilgileri okuyup vize ve final notlarına göre geçme notu ($\text{geçme notu} = \text{vize} * 0,4 + \text{final} * 0,6$) hesaplayıp tüm bilgileri 'gecmenotu.txt' isimli dosyaya yazdıran ve 'gecmenotu.txt' isimli dosyadaki tüm bilgileri listeleyen bir C programı yazınız.

```


EOF girerek veri girisini sonlandiriniz.
isim giriniz>>aaa
numara giriniz>>89
vize notunu giriniz>>78
final notunu giriniz>>90
EOF girerek veri girisini sonlandiriniz.
isim giriniz>>bbb
numara giriniz>>67
vize notunu giriniz>>90
final notunu giriniz>>80
EOF girerek veri girisini sonlandiriniz.
isim giriniz>>ccc
numara giriniz>>87
vize notunu giriniz>>65
final notunu giriniz>>75
EOF girerek veri girisini sonlandiriniz.
isim giriniz>>^Z
numara giriniz>>^Z
vize notunu giriniz>>^Z
final notunu giriniz>>^Z
gecme notlari hesaplan²yor...
Numara      Isim      Vize      Final      Gecme Notu
89          aaa      78       90        85
67          bbb      90       80        84
87          ccc      65       75        71

```



 ogrenci - Not Defteri

Dosya	Düzen	Biçim	Görünüm	Yardım
89	aaa	78	90	
67	bbb	90	80	
87	ccc	65	75	

 gecmenotu - Not Defteri

Dosya	Düzen	Biçim	Görünüm	Yardım
89		aaa	78	90
67		bbb	90	80
87		ccc	65	75

İki adet dosya oluşturuldu!!!

C kodu:

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int vize,final,num,gn;
    char isim[ 30 ];
    FILE *cfPtr,*cfPtr2;

    if ( ( cfPtr = fopen("ogrenci.txt", "w" ) ) == NULL )
        printf( "Dosya acilmadi\n" );
    else {
        printf( " EOF girerek veri girisini sonlandiriniz.\n" );
        printf( "isim giriniz>>" );
        scanf ("%s",isim);
        printf( "numara giriniz>>" );
        scanf ("%d",&num);
        printf( "vize notunu giriniz>>" );
        scanf ("%d",&vize);
        printf( "final notunu giriniz>>" );
        scanf ("%d",&final);
```

```
while ( !feof( stdin ) ) {  
    fprintf( cfPtr, "%d %s %d %d \n", num, isim, vize, final );  
    printf( " EOF girerek veri girisini sonlandiriniz.\n" );  
    printf( "isim giriniz>>" );  
    scanf ("%s",isim);  
    printf( "numara giriniz>>" );  
    scanf ("%d",&num);  
    printf( "vize notunu giriniz>>" );  
    scanf ("%d",&vize);  
    printf( "final notunu giriniz>>" );  
    scanf ("%d",&final);  
}  
fclose( cfPtr );  
}
```

// ogrenci.txt isimli dosya oluřturuldu...

// ogrenci.txt isimli dosyadan veri okunup, geme notları gecmenotu.txt isimli dosyaya yazdırılıyor.

```
if ( ( cfPtr2 = fopen("gecmenotu.txt", "w" ) ) == NULL )
    printf( "Dosya acilmadi\n" );
else {
    if ( ( cfPtr = fopen( "ogrenci.txt", "r" ) ) == NULL )
        printf( "Dosya acilmadi\n" );
    else {
        printf( "gecme notlari hesaplanıyor...\n");
        fscanf( cfPtr, "%d %s %d %d", &num, isim, &vize, &final );

        while ( !feof( cfPtr ) ) {
            gn=(0.4*vize)+(0.6*final);
            fprintf( cfPtr2, "%-13d%-13s%-13d%-13d%-13d\n",num, isim, vize, final, gn );
            fscanf( cfPtr, "%d%s%d%d", &num, isim, &vize, &final );
        }
        fclose( cfPtr );
    }
    fclose( cfPtr2 );
}
```

// gecmenotu.txt isimli dosya ekrana yazdırılıyor...

```
if ( ( cfPtr2 = fopen( "gecmenotu.txt", "r" ) ) == NULL )
    printf( "Dosya acilmadi\n" );
else {
    printf( "%-10s%-10s%-10s%-10s%-10s\n", "Numara", "Isim", "Vize", "Final", "Gecme Notu"
);
    fscanf( cfPtr2, "%d %s %d %d %d", &num, isim, &vize, &final, &gn );
    while ( !feof( cfPtr2 ) ) {
        printf( "%-10d%-10s%-10d%-10d%-10d\n", num, isim, vize, final, gn );
        fscanf( cfPtr2, "%d %s %d %d %d", &num, isim, &vize, &final, &gn );
    }
    fclose( cfPtr2 );
}

getch();
return 0;
}
```

Soru2: İçerisinde öğrenci ismi, numarası, cep telefonu numarası ve doğum tarihini içeren bilgileri saklayan bir dosyayı oluşturan, oluşturulan dosyaya yeni kayıt eklemesi yapan, dosyayı ekrana listeleyen, dosyadaki bir kaydı silen ve dosyadaki bir kaydı düzeltme yapan bir C programı yazınız.

```
Yeni bir dosya için 1 --->
Ekleme yapmak için 2 --->
Listelemek için 3 --->
Kayıt silmek için 4 --->
Kayıt düzeltmek için 5 --->

Lutfen tercihinizi giriniz: _
```


C kodu:

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

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

```
#include <string.h>
```

```
#include <stdlib.h>
```

```
void yeni();
```

```
void ekle();
```

```
void listele();
```

```
void sil();
```

```
void duzelt();
```

```
int main()
```

```
{ int a,b;
```

```
    bas:
```

```
    printf ("Yeni bir dosya icin 1 ---> \nEkleme yapmak icin 2 ---> \nListelemek icin 3 ---> \nKayit silmek icin 4 ---> \nKayit duzeltmek icin 5 ---> \n");
```

```
    printf ("\nLutfen tercihinizi giriniz: ");
```

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

```
    if (a==1) yeni();
```

```
    if (a==2) ekle();
```

```
    if (a==3) listele();
```

```
    if (a==4) sil();
```

```
    if (a==5) duzelt();
```





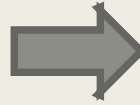
```
printf ("baska islem varsa 1, yoksa 2 giriniz\n");  
scanf ("%d",&b);  
if (b==1) {system("CLS"); goto bas;}  
if (b==2) {  
son:  
    getch();  
    return 0; }  
}
```





```
void yeni()
{ int okulno;
  char isim[30],telno[30],dogum[30];
  FILE *cfPtr;

  if ( ( cfPtr = fopen("ogrenci.txt", "w" ) ) == NULL )
    printf( "Dosya acilmadi\n" );
  else {
    printf( "ogrencinin okul numarasini giriniz.\n" );
    scanf ("%d",&okulno);
    printf( "ogrencinin ismini giriniz.\n" );
    scanf ("%s",isim);
    printf( "ogrencinin tel numarasini giriniz.\n" );
    scanf ("%s",telno);
    printf( "ogrencinin dogum tarihini giriniz.\n" );
    scanf ("%s",dogum);
```



```
while ( !feof( stdin ) ) {
    fprintf( cfPtr, "%d %s %s %s \n", okulno, isim,
telno, dogum );

    printf( " EOF girerek veri girisini sonlandiriniz.\n" )
    printf( "ogrencinin okul numarasini giriniz.\n" );
    scanf ("%d",&okulno);
    printf( "ogrencinin ismini giriniz.\n" );
    scanf ("%s",isim);
    printf( "ogrencinin tel numarasini giriniz.\n" );
    scanf ("%s",telno);
    printf( "ogrencinin dogum tarihini giriniz.\n" );
    scanf ("%s",dogum);
}

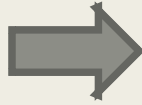
fclose( cfPtr );
}
```





```
void ekle()
{ int okulno;
  char isim[30],telno[30],dogum[30];
  FILE *cfPtr;

  if ( ( cfPtr = fopen("ogrenci.txt", "a" ) ) == NULL )
    printf( "Dosya acilmadi\n" );
  else {
    printf( "ogrencinin okul numarasini giriniz.\n" );
    scanf ("%d",&okulno);
    printf( "ogrencinin ismini giriniz.\n" );
    scanf ("%s",isim);
    printf( "ogrencinin tel numarasini giriniz.\n" );
    scanf ("%s",telno);
    printf( "ogrencinin dogum tarihini giriniz.\n" );
    scanf ("%s",dogum);
```



```
while ( !feof( stdin ) ) {
    fprintf( cfPtr, "%d %s %s %s \n", okulno, isim,
telno, dogum );
    printf( " EOF girerek veri girisini sonlandiriniz.\n" );
    printf( "ogrencinin okul numarasini giriniz.\n" );
    scanf ("%d",&okulno);
    printf( "ogrencinin ismini giriniz.\n" );
    scanf ("%s",isim);
    printf( "ogrencinin tel numarasini giriniz.\n" );
    scanf ("%s",telno);
    printf( "ogrencinin dogum tarihini giriniz.\n" );
    scanf ("%s",dogum);
}

fclose( cfPtr );
}
```





```
void listele()
{ int okulno;
  char isim[30],telno[30],dogum[30];
  FILE *cfPtr;

  if ( ( cfPtr = fopen( "ogrenci.txt", "r" ) ) == NULL )
    printf( "Dosya acilmadi\n" );
  else {
    printf( "%-10s%-13s%-13s%-13s\n", "Numara", "Isim", "Telno", "Dogum T." );
    fscanf( cfPtr, "%d %s %s %s ", &okulno, isim, telno, dogum );

    while ( !feof( cfPtr ) ) {
      printf( "%-10d%-13s%-13s%-13s\n", okulno, isim, telno, dogum);
      fscanf( cfPtr, "%d %s %s %s ", &okulno, isim, telno, dogum);
    }
    printf( "%-10d%-13s%-13s%-13s\n", okulno, isim, telno, dogum);
    fclose( cfPtr );
  }
}
```



// Dosyadaki bütün kayıtlar taranır ve silinmesini istediğimiz kayıt atlanarak geçici bir dosyaya, silinecek kayıt dışındakilerin hepsi yazılır//

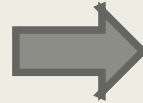


```
fscanf( cfPtr, "%d %s %s %s ", &okulno, isim, telno, dogum );
```

```
void sil()
{ int okulno,silno;
  char isim[30],telno[30],dogum[30];
  FILE *cfPtr;
  FILE *silcfPtr;

  printf("Silmek istediginiz kisinin
numarasini giriniz : \n");
  scanf("%d", &silno);

  cfPtr = fopen("ogrenci.txt","r");
  silcfPtr =
fopen("yeniogrenci.txt","w");
```



```
while ( !feof( cfPtr ) ) {
    if( silno!=okulno ) {
        fprintf( silcfPtr, "%d %s %s %s \n",
            okulno, isim, telno, dogum ); }
    fscanf( cfPtr, "%d %s %s %s ", &okulno, isim, telno,
dogum);
    }
    if( silno!=okulno ) { fprintf( silcfPtr, "%d %s %s %s \n",
        okulno, isim, telno, dogum );}
    fclose( cfPtr );
    fclose( silcfPtr );
```

```
remove("ogrenci.txt"); // Dosya silinir.
```

```
rename("yeniogrenci.txt","ogrenci.txt");
```

```
// geçici dosyanın ismi asıl dosya ismi değiştirilir.
```



```
}
```



// Dosyadaki bütün kayıtlar taranır ve silinmesini istediğimiz kayıt atlanarak geçici bir dosyaya, silinecek kayıt dışındakilerin hepsi yazılır//

```
void duzelt()  
{int okulno,duzno;  
  char isim[30],telno[30],dogum[30];  
  FILE *cfPtr;  
  FILE *duzcfPtr;  
  
  printf("düzeltmek istediginiz kisinin numarasini giriniz : \n");  
  scanf("%d", &duzno);  
  
  cfPtr = fopen("ogrenci.txt","r");  
  duzcfPtr = fopen("yeniogrenci.txt","w");
```





```
fscanf( cfPtr, "%d %s %s %s ", &okulno, isim, telno, dogum );
while ( !feof( cfPtr ) ) {
    if( duzno!=okulno ) {
        fprintf( duzcfPtr, "%d %s %s %s \n",
                okulno, isim, telno, dogum ); }

    if( duzno==okulno ) {
        printf( "ogrencinin okul numarasini giriniz.\n" );
        scanf ("%d",&okulno);
        printf( "ogrencinin ismini giriniz.\n" );
        scanf ("%s",isim);
        printf( "ogrencinin tel numarasini giriniz.\n" );
        scanf ("%s",telno);
        printf( "ogrencinin dogum tarihini giriniz.\n" );
        scanf ("%s",dogum);
        fprintf( duzcfPtr, "%d %s %s %s ", okulno, isim, telno, dogum);}
        fscanf( cfPtr, "%d %s %s %s ", &okulno, isim, telno, dogum );
    }
}
```





```
if( duzno!=okulno ) {  
    fprintf( duzcfPtr, "%d %s %s %s \n", okulno, isim, telno, dogum ); }  
if( duzno==okulno ) {  
    printf( "ogrencinin okul numarasini giriniz.\n" );  
    scanf ("%d",&okulno);  
    printf( "ogrencinin ismini giriniz.\n" );  
    scanf ("%s",isim);  
    printf( "ogrencinin tel numarasini giriniz.\n" );  
    scanf ("%s",telno);  
    printf( "ogrencinin dogum tarihini giriniz.\n" );  
    scanf ("%s",dogum);  
    fprintf( duzcfPtr, "%d %s %s %s ", okulno, isim, telno, dogum);}  
  
fclose( cfPtr );  
    fclose( duzcfPtr );  
  
remove("ogrenci.txt"); // Dosya silinir.  
rename("yeniogrenci.txt","ogrenci.txt"); // geçici dosyanın ismi asıl dosya ismi değiştirilir.  
}
```

Yeni bir dosya icin 1 --->
Ekleme yapmak icin 2 --->
Listelemek icin 3 --->
Kayit silmek icin 4 --->
Kayit duzeltmek icin 5 --->

Lutfen tercihinizi giriniz: 3

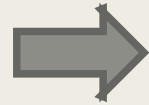
Numara	Isim	Telno	Dogum T.
1145	bbb	53511111	1988
1111	ccc	54444444	2004
1121	aaa	54133333	2002

baska islem varsa 1, yoksa 2 giriniz

1_

```
Yeni bir dosya icin 1 --->
Ekleme yapmak icin 2 --->
Listelemek icin 3 --->
Kayit silmek icin 4 --->
Kayit duzeltmek icin 5 --->

Lutfen tercihinizi giriniz: 4
Silme istediginiz kisinin numarasini giriniz :
1111
baska islem varsa 1, yoksa 2 giriniz
1
```



```
Yeni bir dosya icin 1 --->
Ekleme yapmak icin 2 --->
Listelemek icin 3 --->
Kayit silmek icin 4 --->
Kayit duzeltmek icin 5 --->

Lutfen tercihinizi giriniz: 3
Numara      Isim      Telno      Dogum T.
1145        bbb      53511111  1988
1121        aaa      54133333  2002
baska islem varsa 1, yoksa 2 giriniz
2
```

fread() ve fwrite() Fonksiyonları

--- Verileri dosyaya yazma ve dosyadan okuma işlemlerinde fwrite() ve fread() fonksiyonları da kullanılabilir.

--- Daha önce kullandığımız fprintf() ve fscanf() fonksiyonları dosyadan veri okuma ve dosyaya veri yazma işlemlerinde veriler üzerinde değişim yaparak çalıştığından, fread() ve fwrite() fonksiyonlarını kullanmak daha pratik olabilir. (%d, %s kullanımı!!!)

Yani:

--- fprintf() fonksiyonunu kullanarak bir dosyaya sayı (int) yazarken, sayının dosyanın ASCII metnine çevrilmesi gerekir: `fprintf(fp, "%d", sayi);`

--- fscanf() fonksiyonu ile bir dosyadan bir sayı okurken, sayının fscanf() fonksiyonunun dahili format yapısına çevrilmesi gerekir: `fscanf (fp, "%d", &sayi);`

```
size_t fread ( void      *istenen-bellek,  
                size_t boyut,  
                size_t  no1,  
                FILE *fp);
```

- **fp** ile gösterilen dosyadan okunan boyut yapısında **no1** kadar değeri **istenen-bellek** ile gösterilen belleğe atar.
- Burada, **boyut** ifadesi okunan verinin **byte** olarak değerini, **no1** ifadesi ise kaç adet veri okunduğunu belirler.
- **fread()** fonksiyonu okunan veri sayısını geri verir. Bu değer **0** ise, herhangi bir veri okunmamış demektir. Bu durumda, ya bir hata olmuştur ya da dosya sonu gelmiştir.

```
size_t fwrite (void      *istenen-bellek,  
                size_t boyut,  
                size_t no1,  
                FILE *fp);
```

--- **fwrite()** fonksiyonu, **fread()** fonksiyonunun yaptığı işlemin tam tersini yapar. **fwrite()** fonksiyonu, **istenen-bellek** ile gösterilen bellekte bulunan **boyut** yapısında **no1** kadar değeri **fp** ile gösterilen dosyaya yazar. Burada, **boyut** parametresi yazılan **byte** olarak değerini, **no1** parametresi ise kaç adet veri yazıldığını belirler.

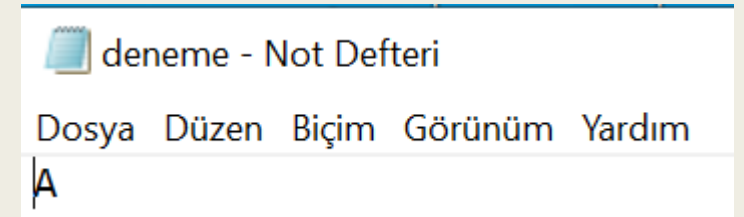
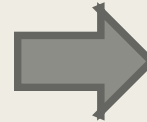
--- **fwrite()** fonksiyonu yazılan veri sayısını geri verir. Sadece bir hata meydana geldiğinde bu değer **no1** değerinden az olur.

Örnek 1:

int veri türünde *x* isimli bir değişken tanımlayıp, bu değere klavyeden bir değer atayınız ve bu değeri fwrite() kullanarak “deneme.txt” dosyasına yazdırınız.

Daha sonra dosyayı tekrar okuma modunda açınız, fread() ile okuma yaparak, int veri türünde *y* isimli değişkene okuduğunuz değeri aktarınız ve *y* değişkeninin değerini ekrana yazdırınız.

```
x degerini giriniz: 65  
y degeri= 65  
-----  
Process exited after 5.03 seconds with return value 0  
Press any key to continue . . .
```



ASCII kod → '65=A'

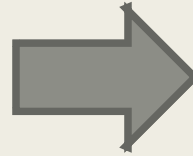
C kodu:

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

```
int main()
{
FILE *fp;
int x,y;
printf("x degerini giriniz: ");
scanf("%d",&x);

fp = fopen ("deneme.txt", "w");
if (fp == NULL)
{
printf("Dosya olusturulamadi...");
exit(1);
}

fwrite(&x, sizeof(int), 1, fp);
fclose(fp);
```



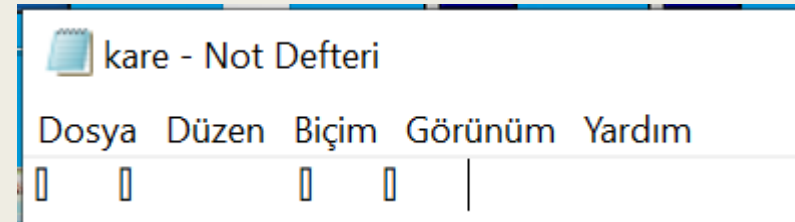
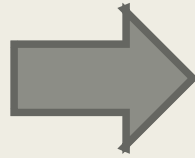
```
fp = fopen ("deneme.txt", "r");
fread(&y, sizeof(int), 1, fp);

printf("y degeri= %d",y);
fclose(fp);
getch();
return 0;
}
```


Örnek 2:

1 den 5 e kadar olan sayıların karelerini önce bir diziye daha sonrada dizi elemanlarını tek tek 'kare.tex' isimli bir dosyaya yazdıran ve daha sonra dosyadan bu elemanları okuyup ekrana yazdıran C programını fwrite ve fread komutları ile yapınız.

```
1. deger= 1  
2. deger= 4  
3. deger= 9  
4. deger= 16  
5. deger= 25
```



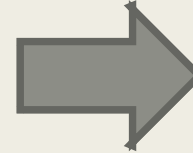
C kodu:

```
#include <stdio.h>
#include <stdlib.h>
int main ()
{
    int dizi[5], i, y;
    FILE *fp;

    for (i = 0; i < 5; i++)
        { dizi[i] = (i+1)*(i+1); }

    fp = fopen("kare.txt", "w");

    if (fp == NULL)
    {
        printf("Dosya olusturulamadi...");
        exit(1);
    }
    for (i = 0; i < 5; i++)
    {
        fwrite(&dizi[i], sizeof(int), 1, fp);
    }
    fclose(fp);
```



```
fp = fopen("kare.txt", "r");

if (fp == NULL)
{
    printf("Dosya olusturulamadi...");
    exit(1);
}

for (i = 0; i < 5; i++)
{
    fread(&y, sizeof(int), 1, fp);
    printf("%d. deger= %d\n", i+1, y);
}
fclose(fp);
return 0;
}
```

Örnek 3: 2. Örnekteki dizi direkt olarak dosyaya yazdıran bir C programı yazınız.

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

```
int main ()
```

```
{
```

```
    int dizi[5],oku[5], i, y;
```

```
    FILE *fp;
```

```
    for (i = 0; i < 5; i++)
```

```
    { dizi[i] = (i+1)*(i+1); }
```

```
    fp = fopen("kare.txt", "w");
```

```
    if (fp == NULL)
```

```
    {
```

```
        printf("Dosya olusturulamadi...");
```

```
        exit(1);
```

```
    }
```

```
    fwrite(dizi, sizeof(dizi), 1, fp);
```

```
    fclose(fp);
```

```
    fp = fopen("kare.txt", "r");
```

```
    if (fp == NULL)
```

```
    {
```

```
        printf("Dosya olusturulamadi...");
```

```
        exit(1);
```

```
    }
```

```
    fread(oku, sizeof(oku), 1, fp);
```

```
    for (i = 0; i < 5; i++)
```

```
    {
```

```
        printf("%d. deger= %d\n",i+1,oku[i]);
```

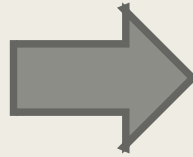
```
    }
```

```
    fclose(fp);
```

```
    return 0;
```

```
}
```

```
1. deger= 1
2. deger= 4
3. deger= 9
4. deger= 16
5. deger= 25
```



rewind() Fonksiyonu

--- Bir dosyaya yazma işlemini tamamladıktan sonra, dosyayı tekrar okumak istediğimizde dosyayı kapatır ve dosyayı okuma modunda tekrar açarız.

```
fopen("kare.txt", "w");
```

```
fclose(fp);
```

```
fopen("kare.txt", "r");
```

```
fclose(fp);
```

--- Dosya açma kapatma işlemi yerine, bir dosyanın aktif konumunu dosya başına almak için `rewind()` fonksiyonunu kullanılır.

--- `fp` parametresi ile gösterilen dosyanın aktif konum göstergesini dosyanın başına alır.

```
fopen("kare.txt", "w+");  
rewind(fp);  
fclose(fp);
```

--- Bu fonksiyon herhangi bir değer geri vermez. Çünkü, başarılı bir şekilde açılan dosyanın aktif konumu başa alınmaktadır.

Örnek 4: 3. Örnekteki C kodunu rewind kullanarak revize ediniz.

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

```
#include <stdlib.h>
```

```
int main ()
```

```
{
```

```
    int dizi[5],oku[5], i, y;
```

```
    FILE *fp;
```

```
    for (i = 0; i < 5; i++)
```

```
    { dizi[i] = (i+1)*(i+1);}
```

```
    fp = fopen("kare.txt", "w+");
```

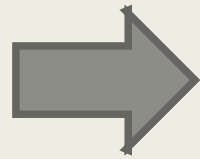
```
    if (fp == NULL)
```

```
    {
```

```
        printf("Dosya olusturulamadi...");
```

```
        exit(1);
```

```
    }
```



```
        fwrite(dizi, sizeof(dizi), 1, fp);
```

```
        rewind(fp);
```

```
        fread(oku, sizeof(oku), 1, fp);
```

```
        for (i = 0; i < 5; i++)
```

```
        {
```

```
            printf("%d. deger= %d\n",i+1,oku[i]);
```

```
        }
```

```
        fclose(fp);
```

```
        return 0;
```

```
    }
```

Örnek 5: İçerinde hesap no, isim ve borç bilgileri bulunan yhesap isimli bir struct oluşturunuz. Klavyeden girilen verileri "musteri.txt" isimli dosyaya yazdıran daha sonra bu bilgileri listeleyen bir C programı yazınız. Müşteri girişi hesap no, sıfır girilinceye kadar devam etmektedir.

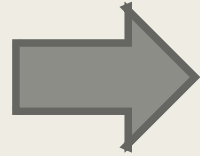
```
-----HESAP VERI GIRISI -----  
Hesap No: 235  
Isim: aaa  
Borc: 4000  
  
Hesap No: 112  
Isim: bbb  
Borc: 5000  
  
Hesap No: 368  
Isim: ccc  
Borc: 2500  
  
Hesap No: 0  
  
-----HESAP LISTESI-----  
Hesap No: 235  
Isim: aaa  
Borc: 4000.000000  
  
Hesap No: 112  
Isim: bbb  
Borc: 5000.000000  
  
Hesap No: 368  
Isim: ccc  
Borc: 2500.000000
```

C kodu:

```
#include <stdio.h>
#include <stdlib.h>
typedef struct
{
    int no;
    char isim[100];
    float borc;
} yhesap;

int main()
{
    FILE *fp;
    yhesap hesap1, hesap2;

    if ((fp=fopen ("musteri.txt", "w+")) == NULL)
    {
        printf("Dosya açilamadi!\n");
        exit(1);
    }
```



```
printf("\n-----HESAP VERI GIRISI ----- \n");
while (1==1)
{
    printf("Hesap No: ");
    scanf("%d",&hesap1.no);

    if (hesap1.no == 0) break;

    printf("Isim: ");
    scanf("%s", hesap1.isim);
    printf("Borc: ");
    scanf("%f",&hesap1.borc);
    if (fwrite(&hesap1, sizeof(hesap1), 1, fp) != 1)
    {
        printf("Yazma hatasi!\n");
        exit(1);
    }
    printf("\n");
}
```





rewind(fp);

```
printf("\n-----HESAP LISTESI-----\n");
while(fread(&hesap2, sizeof(hesap2), 1, fp)==1)
{
    printf("Hesap No: %d\n", hesap2.no);
    printf("Isim: %s\n", hesap2.isim);
    printf("Borc: %f\n", hesap2.borc);
    printf("\n");
}

return 0;
}
```

fseek() Fonksiyonu

--- Şu ana kadarki dosya işlemlerinde dosyadan yaptığımız okuma işlemlerini dosyanın başından sonuna doğru bir sıra ile yaptık.

--- Bu tipte dosya erişimi **sıralı erişim** olarak ifade edilir.

--- Bunun yanında, fseek() fonksiyonunu kullanarak dosyaların herhangi bir yerindeki bilgi okuması yapılabilir.

--- Bu tipte dosya erişimi **rastgele erişim** olarak ifade edilir.

Kullanım Şekli:

```
int fseek (FILE *fp, int ara, int yer);
```

--- fp parametresi işlem yapılan dosyayı,

--- ara parametresi yer parametresinin gösterdiği değerin tanımladığı dosya konumundan, işlem yapılmak istenen yerin byte olarak uzaklığını verir.

--- yer parametresi dosyada arama işleminin başlayacağı yeri gösterir. 3 tane değeri vardır.

--- yer parametresinin alabileceği değerler aşağıdaki gibidir:

- SEEK_SET (0) Aramayı dosya başından başlatır.
- SEEK_CUR (1) Aramayı aktif konumdan başlatır.
- SEEK_END (2) Aramayı dosya sonundan başlatır.

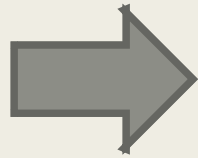
Örnek 6:

İçerinde hesap no, isim ve borç bilgileri bulunan yhesap isimli bir struct oluşturup, "musteri.txt" isimli dosyaya hesap no değeri 0 olana kadar hesap bilgilerini yazdırınız.

--- hesap no'ya göre dosyanın konumlanmasını sağlayıp, dosyaya yazma işlemi için fwrite() fonksiyonunu kullanınız. (konumlanma fseek() fonksiyonu ile yapılacaktır.)

--- En son kayda kadar olan hesap bilgilerini ekrana yazdıran bir C programı yazınız.

```
-----HESAP VERI GIRISI-----  
Hesap No: 3  
Isim: aaa  
Borc: 500  
  
Hesap No: 10  
Isim: bbb  
Borc: 700  
  
Hesap No: 7  
Isim: ccc  
Borc: 250  
  
Hesap No: 0
```



```
-----HESAP LISTESI-----  
Hesap No:0  
Isim:  
Borc:0.000000  
  
Hesap No:0  
Isim:  
Borc:0.000000  
  
Hesap No:3  
Isim:aaa  
Borc:500.000000
```

```
Hesap No:0  
Isim:  
Borc:0.000000
```

```
Hesap No:0  
Isim:  
Borc:0.000000
```

```
Hesap No:0  
Isim:  
Borc:0.000000
```

```
Hesap No:7  
Isim:ccc  
Borc:250.000000
```

```
Hesap No:0  
Isim:  
Borc:0.000000
```

```
Hesap No:0  
Isim:  
Borc:0.000000
```

```
Hesap No:10  
Isim:bbb  
Borc:700.000000
```

C kodu:

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

```
typedef struct
{
    int no;
    char isim[100];
    float borc;
} yhesap;
```

```
int main()
{
    FILE *fp;
    yhesap hesap1, hesap2;
```

```
    if ((fp=fopen ("musteri.txt", "w+")) == NULL)
    {
        printf("Dosya açilamadi!\n");
        exit(1);
    }
```





```
printf("\n-----HESAP VERI GIRISI-----\n");
while (1)
{
    printf("Hesap No: ");
    scanf("%d",&hesap1.no);

    if (hesap1.no == 0) break;

    printf("Isim: ");
    scanf("%s", hesap1.isim);
    printf("Borc: ");
    scanf("%f",&hesap1.borc);

    fseek(fp, (hesap1.no - 1) * sizeof(hesap1), SEEK_SET); // konumlama

    if (fwrite(&hesap1, sizeof(hesap1), 1, fp) != 1)
    {
        printf("Yazma hatasi!\n");
        exit(1);
    }
    printf("\n");
}
```

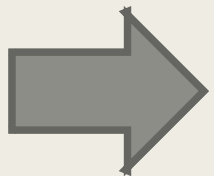




```
rewind(fp);

printf("\n-----HESAP LISTESI-----\n");
while(fread(&hesap2, sizeof(hesap2), 1, fp))
{
    printf("Hesap No:%d\n", hesap2.no);
    printf("Isim:%s\n", hesap2.isim);
    printf("Borc:%f\n", hesap2.borc);
    printf("\n");
}

return 0;
}
```



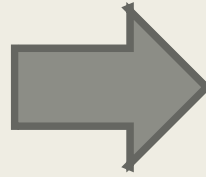
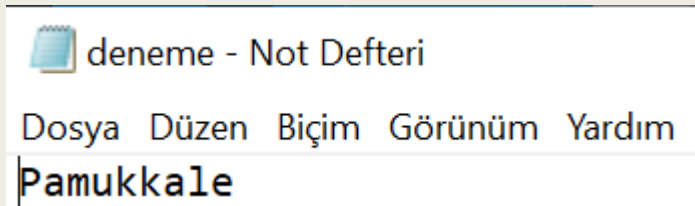
Yazma işleminden hemen önce dosya konumlandırma işlemi yapılır.

```
fseek(fp, (hesap1.no - 1) * sizeof(hesap1), SEEK_SET); // konumlama
```


Örnek 7:

--- Pamukkale kelimesini " deneme.txt " isimli dosyaya yazdıran, ve daha sonra dosya konumuna en başa aldırıp konumu ve bu konumdan sonraki tüm ifadeleri yazdıran.

--- Son olarak en sona gelen dosya konumunu 4 birim sola kaydırıp yeni konumu ekrana yazdıran ve yeni konumdan sonraki tüm karakterleri ekrana yazdıran bir C programı yazınız.



```
Dosyanın konumu: 0 Karakter dizisi: Pamukkale
Dosyanın yeni konumu: 5 Karakter dizisi: kale
-----
```

C kodu:

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

```
#include <stdlib.h>
```

```
int main()
```

```
{
```

```
    FILE *fp;
```

```
    fpos_t pos;
```

```
    char A[10]="Pamukkale";
```

```
    if ((fp = fopen ("deneme.txt", "w+")) == NULL) {
```

```
        printf("Dosya acilamadi...");
```

```
        exit(1);
```

```
    }
```

```
    fprintf(fp,"%s",A);
```

```
    fseek(fp, 0, SEEK_SET); // Dosya konumunu en başa alınıyor
```

```
    fgetpos(fp, &pos); // Pozisyonu belirliyor
```

```
    printf("Dosyanin konumu: %ld ", pos); // Pozisyonu yazdırılıyor
```

```
    fscanf(fp,"%s",A);
```

```
    printf("Karakter dizisi: %s\n", A);
```





```
fseek(fp, -4, SEEK_CUR); // Pozisyon en sondan 4 birim solda
fgetpos(fp, &pos); // Pozisyonu belirliyor
printf("Dosyanin yeni konumu: %ld ", pos); // Yeni pozisyonu yazdırılıyor
fscanf(fp,"%s",A); // Yeni pozisyondan başlayıp yazdırma yapıyor
printf("Karakter dizisi: %s\n",A);

fclose(fp);

return 0;
}
```

Kaynaklar

- C: How to Program Third Edition Harvey M. Deitel ; Paul J. Deitel.
- C Programlama Dili Dr. Rıfat Çölkesen Papatya Yayıncılık.
- Problem Solving and Program Design in C, 7/E Jeri R. Hanly; Elliot B. Koffman.
- C Programlama dili; İbrahim Güney; Nobel Yayıncılık.
- Algoritma Geliştirme ve Programlamaya Giriş, Fahri Vatansever, Seçkin yayıncılık
- C Programlama Ders Notları, A. Kadir YALDIR, Pamukkale Üniversitesi ders notları.