



**T.C.**  
**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**SİSTEM PROGRAMLAMA ÖDEV RAPORU**

**SENA NUR ERDEM**

**G201210033**

**SAKARYA**

**Mayıs, 2024**

Sistem Programlama Dersi

## Programın derlenip çalıştırılması

```
SistemProgramlamaProje — -zsh — 80x24
[senanurerdem@Sena-MacBook-Air ~ % cd desktop ]
[senanurerdem@Sena-MacBook-Air desktop % cd SistemProgramlamaProje ]
[senanurerdem@Sena-MacBook-Air SistemProgramlamaProje % make ]
cc -I "./include" -c ./src/main.c -o ./obj/main.o
cc -I "./include" -c ./src/fields.c -o ./obj/fields.o
cc -I "./include" -c ./src/jval.c -o ./obj/jval.o
cc -I "./include" -c ./src/dllist.c -o ./obj/dllist.o
cc -I "./include" -c ./src/jrb.c -o ./obj/jrb.o
cc ./obj/fields.o ./obj/main.o ./obj/dllist.o ./obj/jval.o ./obj/jrb.o -o ./bin/program
ar ru lib/libfdr.a obj/jval.o obj/jrb.o obj/dllist.o obj/fields.o
ranlib lib/libfdr.a
[senanurerdem@Sena-MacBook-Air SistemProgramlamaProje % bin/program giris ]
kullanım: bin/program giris_dosya_ismi cikis_dosya_ismi
[senanurerdem@Sena-MacBook-Air SistemProgramlamaProje % bin/program giris cikis ]
Giris ve cikis dosyalari .dat uzantili olmalidir!
[senanurerdem@Sena-MacBook-Air SistemProgramlamaProje % bin/program giris.dat cikis ]
Giris ve cikis dosyalari .dat uzantili olmalidir!
[senanurerdem@Sena-MacBook-Air SistemProgramlamaProje % bin/program giris.dat cikis.dat ]
senanurerdem@Sena-MacBook-Air SistemProgramlamaProje % █
```

‘make’ komutu ile program derlenir.

Hem giriş hem de çıkış dosyası komut satırından parametre olarak verilmelidir.

Parametrelerden biri eksik olursa uyarı verir.

Parametre olarak verilen giriş ve çıkış dosyalarının uzantısı .dat değilse uyarı verir.

Ödev dokümanında verilen örnek giriş dosyası çalıştırıldığına yine dökümandaki gibi doğru çıktı alındı.

```
0x giris.dat x
0x giris.dat
1 yaz: 10 a 1 \b 8 k 4 r 6 u
2 sil: 4 a
3 sonagit:
4 yaz: 3 b 4 c 1 \n
5 yaz: 2 s
6 sonagit:
7 dur:

0x cikis.dat x
0x cikis.dat
1 aaaaaa kkkkkkkrrrruuuuuubbbccccc
2 ss
```

Örnekteki sil komutundan sonra sonagit komutu olmasaydı:

```
0x giris.dat x
0x giris.dat
1 yaz: 10 a 1 \b 8 k 4 r 6 u
2 sil: 4 a
3 yaz: 3 b 4 c 1 \n
4 yaz: 2 s
5 sonagit:
6 dur:

0x cikis.dat x
0x cikis.dat
1 aaaaaabbbccccc
2 ss kkkkkkkrrrruuuuuuu
```

Kendi denediğim diğer örnekler:

Sil komutunun birden fazla operand içermesi halinde

```
0x giris.dat ×
0x giris.dat
1 yaz: 10 a 1 \b 8 k 4 r 6 u
2 sil: 4 a 6 k
3 yaz: 3 b 4 c 1 \n
4 yaz: 2 s
5 sonagit:
6 dur:

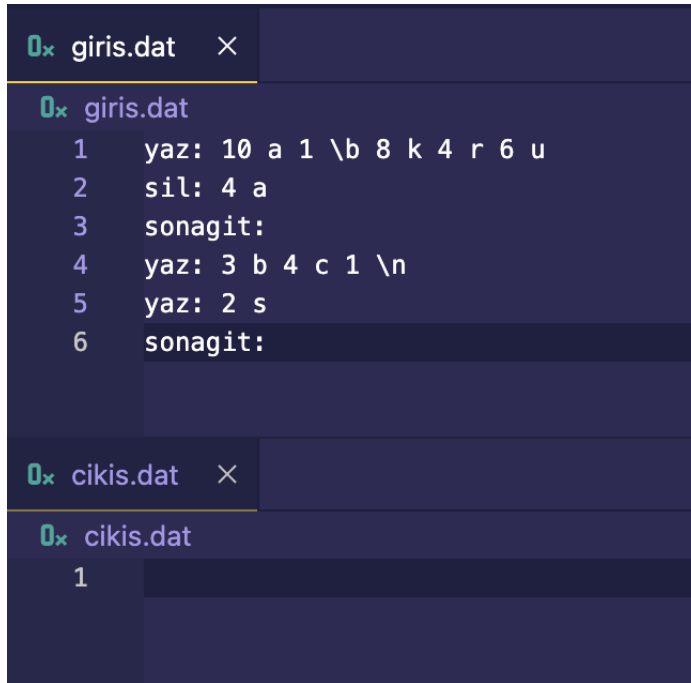
0x cikis.dat ×
0x cikis.dat
1 aaaaaa kkbbbcccc
2 ssrrrrrruuuuuu
```

Arka arkaya iki sil komutu olması halinde

```
0x giris.dat ×
0x giris.dat
1 yaz: 10 a 1 \b 8 k 4 r 6 u
2 sil: 4 a
3 sil: 6 k
4 sonagit:
5 yaz: 3 b 4 c 1 \n
6 yaz: 2 s
7 sonagit:
8 dur:

0x giris.dat 0x cikis.dat ×
0x cikis.dat
1 aaaaaa kkrrrrrruuuuuubbbcccc
2 ss
```

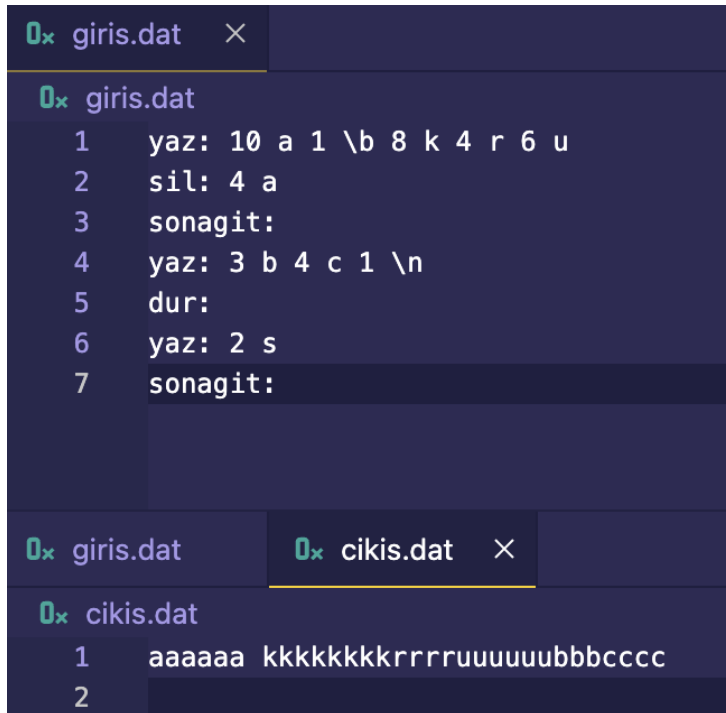
Dur komutu olmazsa program çıktı vermez. Çünkü dosyaya yazma işlemini dur komutu yapar.



```
0x giris.dat x
0x giris.dat
1   yaz: 10 a 1 \b 8 k 4 r 6 u
2   sil: 4 a
3   sonagit:
4   yaz: 3 b 4 c 1 \n
5   yaz: 2 s
6   sonagit:

0x cikis.dat x
0x cikis.dat
1
```

Eğer dur komutundan sonra başka komutlar gelirse bunlar işlenmez. Çünkü dur komutu programdan çıkış yapar.



```
0x giris.dat x
0x giris.dat
1   yaz: 10 a 1 \b 8 k 4 r 6 u
2   sil: 4 a
3   sonagit:
4   yaz: 3 b 4 c 1 \n
5   dur:
6   yaz: 2 s
7   sonagit:

0x giris.dat 0x cikis.dat x
0x cikis.dat
1   aaaaaa kkkkkkkrrrruuuuuubbbccccc
2
```

Eğer sil yaz dur ve sonagit komutlarından başka bir komut ve operandları gelirse bunlar atlanır. Program çökmez.

```
0x giris.dat x
0x giris.dat
1 yaz: 10 a 1 \b 8 k 4 r 6 u
2 sil: 4 a
3 sonagit:
4 yaz: 3 b 4 c 1 \n
5 gecersiz: 8 p
6 yaz: 2 s
7 sonagit:
8 dur:

0x giris.dat 0x cikis.dat x
0x cikis.dat
1 aaaaaa kkkkkkkrrrruuuuuubbbcccc
2 ss
```

Veya sil yaz komutlarına geçersiz bir operand gelirse ( yani bir sayı bir harf ikilisi dışında) bu geçeriz operandlar atlanır. Program çökmez.

```
0x giris.dat x
0x giris.dat
1 yaz: 10 a 1 \b 8 k 4 r 6 u
2 sil: 4 a 1 6 k
3 sonagit:
4 yaz: 3 b p 4 c 1 \n
5 yaz: 2 s
6 sonagit:
7 dur:

0x cikis.dat x
0x cikis.dat
1 aaaaaa krrrruuuuuubbbcccc
2 ss
```

```
0x giris.dat ×
0x giris.dat
1 yaz: 10 a 1 \b 8 k 4 r 6 u
2 sil: 4 a 1 6 k
3 sonagit:
4 yaz: 3 b p 4 c 1 \n
5 yaz: 2 s
6 sonagit:
7 dur:

0x cikis.dat ×
0x cikis.dat
1 aaaaaa kkrrrruuuuuubbbccccc
2 ss
```

```
0x giris.dat ×
0x giris.dat
1 yaz: 10 a 1 \b 8 k 4 r 6 u
2 sil: 4 a f y w 6 k
3 sonagit:
4 yaz: 3 b d 8 4 c 1 \n
5 yaz: 2 s
6 sonagit:
7 dur:

0x giris.dat 0x cikis.dat ×
0x cikis.dat
1 aaaaaa kkrrrruuuuuubbbccccc
2 ss
```

Bu örneklerde görüldüğü gibi geçersiz operandlar geçerli bir parametre gelene kadar atlanır ve program çökmez.