

## Proiect 2.1 – mkdir și rmdir

Avem în memorie un șir de numere de tip WORD. Se dau două numere (deasemenea în memorie, nu trebuie citite de la tastatură) a și b. Toate aparițiile lui a trebuie înlocuite cu b și toate aparițiile lui b trebuie înlocuite cu a.

Șirul rezultat se va afișa pe ecran. Atenție: șirul trebuie să se modifice și în memorie, nu e suficient să se afișeze rezultatul.

Exemplu:

```
sir DW 100, 24, 75, 24, 86, 91, 75, 24
lungime_sir EQU ($-sir)/2
a DW 75
b DW 24
```

Programul va trebui să afișeze: 100 75 24 75 86 91 24 75

## Proiect 2.2 – mkdir și rmdir

Avem în memorie un șir de numere de tip BYTE. Se mai dă un număr x (deasemenea în memorie, nu trebuie citit de la tastatură). În cazul în care x apare deja în șir, se va afișa un mesaj corespunzător. Altfel, se va căuta prima apariție a lui 0 și se va înlocui cu x.

La final, șirul rezultat se va afișa pe ecran. Atenție: șirul trebuie să se modifice și în memorie, nu e suficient să se afișeze rezultatul.

Exemplu:

```
sir DB 10, 15, 7, 0, 0, 98, 0, 1, 4
lungime_sir EQU $-sir
x DB 42
```

Programul va trebui să afișeze: 10 15 7 42 0 98 0 1 4

În cazul în care în loc de 42, x era 98, programul ar fi afișat "98 exista deja in sir".

### Proiect 3 – rmtree

Avem în memorie un șir de numere de tip DWORD. Se mai dă un număr x (deasemenea în memorie, nu trebuie citit de la tastatură). Să se șteargă din șir (prin înlocuire cu 0), toate elementele din intervalul  $[x-5, x+5]$ .

La final, șirul rezultat se va afișa pe ecran. Atenție: șirul trebuie să se modifice și în memorie, nu e suficient să se afișeze rezultatul.

Exemplu:

```
sir DD 20, 24, 12, 31, 42, 19, 65, 87, 21
lungime_sir (EQU $-sir)/4
x DB 22
```

Programul va trebui să afișeze: 0 0 12 31 42 0 65 87 0

### Proiect 5.1 – copy sau defrag

Se dă în memorie un număr x, de tip DWORD. Să se afișeze pe ecran lungimea celei mai lungi secvențe de biți de 1 consecutivi.

Exemplu:

```
x DD 111001b
se va afișa 3.
x DD 11000111110b
se va afișa 5
```

### Proiect 5.2 – copy sau defrag

Avem în memorie un șir de numere de tip BYTE. Să se afișeze pe ecran lungimea celei mai lungi secvențe de 0-uri consecutive.

Exemplu:

```
sir DB 10, 15, 0, 0, 16, 0, 0, 0, 1, 24
lungime_sir EQU $-sir
```

Se va afișa 3.

### Proiect 6.1 – rm

Avem în memorie un șir de numere de tip BYTE. Se mai dă un număr x (deasemenea în memorie, nu trebuie citit de la tastatură). Să se șteargă (prin înlocuire cu 0), toate numerele pare mai mici strict decât x.

La final, șirul rezultat se va afișa pe ecran. Atenție: șirul trebuie să se modifice și în memorie, nu e suficient să se afișeze rezultatul.

Exemplu:

```
sir DB 10, 15, 7, 28, 16, 98, 4, 20, 1, 24
lungime_sir EQU $-sir
x DB 20
```

Programul va trebui să afișeze: 0 15 7 28 0 98 0 20 1 24

### Proiect 6.2 – rm

Avem în memorie un șir de numere de tip WORD. Se mai dă un număr x (deasemenea în memorie, nu trebuie citit de la tastatură). Să se șteargă (prin înlocuire cu 0), primul număr impar, mai mare strict decât x.

La final, șirul rezultat se va afișa pe ecran. Atenție: șirul trebuie să se modifice și în memorie, nu e suficient să se afișeze rezultatul.

Exemplu:

```
sir DW 10, 15, 7, 29, 16, 98, 4, 35, 1, 24, 41, 59
lungime_sir EQU ($-sir)/2
x DW 29
```

Programul va trebui să afișeze: 10 15 7 29 16 98 4 0 1 24 41 59

### Proiect 7.1 – show

Avem în memorie un șir de caracter, terminat în 0 (un string). Să se afișeze string-ul pe ecran, câte un caracter pe fiecare linie.

Exemplu:

```
    sir DB "Sirul meu", 0
```

Programul va trebui să afișeze:

```
S
i
r
u
l

m
e
u
```

### Proiect 7.2 – show

Avem în memorie un șir de caracter, terminat în 0 (un string). Să se afișeze pe ecran codul ASCII pentru fiecărui caracter.

Exemplu:

```
    sir DB "Sirul meu", 0
```

Programul va trebui să afișeze: 83 105 114 117 108 32 109 101 117

### Proiect 8.1 – compress

Avem în memorie un șir de numere de tip WORD. Să se înlocuiască fiecare a 2-a apariție consecutivă a unui element cu 0.

La final, șirul rezultat se va afișa pe ecran. Atenție: șirul trebuie să se modifice și în memorie, nu e suficient să se afișeze rezultatul.

Exemplu:

```
sir DW 10, 11, 11, 44, 23, 23, 23, 10, 10, 9
lungime_sir EQU ($-sir)/2
```

Programul va trebui să afișeze: 10 11 0 44 23 0 23 10 0 9

### Proiect 8.2 – encrypt

Avem în memorie un șir de numere de tip DWORD. Să se inverseze între ele fiecare grup de 2 elemente. Dacă lungimea e impară, ultimul element va rămâne nemodificat.

La final, șirul rezultat se va afișa pe ecran. Atenție: șirul trebuie să se modifice și în memorie, nu e suficient să se afișeze rezultatul.

Exemplu:

```
sir DD 4, 8, 125, 76, 89, 34, 18, 12, 10
lungime_sir EQU ($-sir)/4
```

Programul va trebui să afișeze: 8 4 76 125 34 89 12 18 10