

FIȘA DE LABORATOR 5

Tipul *struct*

1. Într-un text cu cel mult 100 de caractere cuvintele sunt formate numai din litere mari și mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat, apoi un număr natural, n ($n \in [1, 10^2]$), și afișează pe ecran, pe linii separate, cuvintele din text care au exact n litere. Cuvintele sunt afișate în ordinea apariției lor în text, iar dacă nu există niciun astfel de cuvânt, se afișează pe ecran mesajul **nu exista**.

Exemplu: dacă se citește textul `Fat Frumos este cel mai viteaz`

iar $n=6$, se afișează pe ecran:

```
Frumos  
viteaz
```

2. Un text are cel mult 100 de caractere, iar cuvintele sale sunt formate numai din litere mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un număr natural n ($n \in [1, 10^2]$), apoi un text de tipul precizat mai sus, și afișează pe ecran cuvintele acestuia, pe rânduri separate, astfel încât primele poziții să fie ocupate de mulțimea formată de cele care au cel puțin n litere, iar următoarele poziții, în continuarea acestora, să fie ocupate de mulțimea celorlalte cuvinte.

Cuvintele din aceeași mulțime sunt afișate într-o ordine oarecare, iar dacă una dintre cele două mulțimi este vidă, se afișează pe ecran doar mesajul **nu exista**.

Exemplu: pentru $n=5$ și textul `e1 mergea tot spre aleea pietruita` datele afișate pot fi cele alăturate.

```
mergea  
aleea  
pietruita  
e1  
tot  
spre
```

3. Într-un text cu cel mult 100 de caractere, cuvintele sunt formate din litere mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul menționat și afișează pe ecran, pe linii separate, toate cuvintele sale pentru care numărul de vocale este strict mai mic decât numărul de consoane. Dacă nu există niciun astfel de cuvânt, se afișează pe ecran mesajul **nu exista**. Se consideră vocale literele din mulțimea a, e, i, o, u .

Exemplu: pentru textul `ei au plantat tamarix ea a adus iasomie` se afișează pe ecran, nu neapărat în această ordine, cuvintele alăturate.

```
plantat  
tamarix
```

4. Un cuvânt este **prefix** al unui alt cuvânt dacă se obține din acesta, prin eliminarea ultimelor sale litere. Scrieți un program C/C++ care citește de la tastatură un număr natural n ($n \in [2, 20]$) și apoi n cuvinte distincte, fiecare fiind format din cel mult 20 de caractere, numai litere mici ale alfabetului englez.

La introducerea datelor, după fiecare cuvânt se tastează Enter. Programul afișează pe ecran, separate prin câte un spațiu, cuvintele care îl au drept prefix pe ultimul cuvânt citit. Dacă nu există astfel de cuvinte, se afișează pe ecran mesajul **nu exista**.

Exemplu: dacă $n=6$ și se citesc cuvintele alăturate, pe ecran se afișează
`raita raid raion`

```
raita  
grai  
raid  
raion  
straie  
rai
```

5. Un cuvânt este **sufix** al unui alt cuvânt dacă se obține din acesta, prin eliminarea primelor sale litere. Scrieți un program C/C++ care citește de la tastatură două numere naturale n și k ($n \in [2, 20]$, $k \in [1, n]$) și apoi n cuvinte distincte, fiecare fiind format din cel mult 20 de caractere, numai litere mici ale alfabetului englez.

La introducerea datelor, după fiecare cuvânt se tastează Enter. Programul afișează pe ecran, separate prin câte un spațiu, cuvintele care îl au drept sufix pe al k -lea cuvânt citit, ca în exemplu. Dacă nu există astfel de cuvinte, se afișează pe ecran mesajul **nu exista**.

Exemplu: dacă $n=7$, $k=3$ și se citesc cuvintele alăturate, pe ecran se afișează
`paratirisi hiritisi`

```
isihast  
paratirisi  
isi  
meremetisire  
acolisitor  
hiritisi  
paraponisit
```

6. Se citește de la tastatură un număr natural n ($1 < n < 30$), apoi se citesc de la tastatură n fracții date prin numărător și numitor (numere naturale nenule). Să se determine și să se afișeze pe ecran, separate prin câte un spațiu, numărătorul respectiv numitorul celei mai mari fracții date, adusă la forma ireductibilă.

Exemplu:

pentru $n=4$ și fracțiile

8 20

5 12

21 12

2 6

se afișează 7 4

7. Se citește de la tastatură un număr natural nenul de cel mult 6 cifre. Să se descompună numărul în factori primi, memorând rezultatul sub forma unui vector de structuri cu două câmpuri: un câmp pentru factorul prim și un câmp pentru puterea la care apare factorul în descompunere.

Exemplu:

pentru numărul 90

se afișează

2 1

2 2

5 1

8. Se citesc de la tastatură două numere raționale pozitive. Să se determine și să se afișeze pe ecran rezultatele următoarelor operații matematice: adunare, scădere, înmulțire și împărțire.

Exemplu:

pentru $a=5.8$ și $b=6.7$

se afișează

1.48214

-0.232143

0.535714

0.729167

9. Se citesc de la tastatură două numere mari, fiecare număr având cel mult 1000 de cifre. Să se determine și să se afișeze pe ecran suma celor două numere.

Exemplu:

pentru $a=4178302154789$ și $b=698472510326457778$

se afișează 698476688628612567

10. O matrice $A(n, m)$ cu elemente întregi se numește rară dacă majoritatea elementelor sale sunt egale cu 0. O matrice rară $A(n, m)$ având k elemente nenule poate fi memorată folosind un șir X conținând k triplete de forma (linie, coloană, valoare) corespunzătoare valorilor nenule ale matricei – fără a folosi un tablou bidimensional. Să se scrie un program care citește de la tastatură valorile n , m și două matrice rare $A(n, m)$ și $B(n, m)$, calculează sub forma unei matrice rare suma $C(n, m)$ a celor două matrice A și B și afișează sub forma unui tablou bidimensional matricea $C(n, m)$. Citirea unei matrice se va face prin citirea numărului de linii (n), numărului de coloane (m) și citirea repetată a unor triplete (linie, coloană, valoare) – corespunzătoare valorilor nenule din matrice, până la citirea tripletului $(-1, -1, -1)$. În cazul citirii mai multor triplete cu aceeași linie și coloană, se ia în considerare doar primul triplet citit.

Exemplu:

pentru $n=m=3$ și matricea A de forma:

0 5 2

0 2 0

2 0 3

se va memora sub forma șirului X de forma:

$X = ((1, 2, 5), (1, 3, 2), (2, 2, 2), (3, 1, 2), (3, 3, 3))$

(Universitatea Babeș-Bolyai Cluj-Napoca,
Facultatea de Matematică și Informatică,
Concurs mate-info aprilie 2015, Subiectul III)