

Problemele rămase din seminarul 4

3. Se citesc din fișierul “puncte.in” informații despre puncte în plan: cele două coordonate (numere întregi) și eticheta (care poate fi un șir ce conține și spații). Mai exact, structura fișierului este următoarea: pe fiecare linie a fișierului sunt date informațiile despre un punct: două coordonate întregi și eticheta punctului.

```
1 2 punctul 1
1 3 punctul 2
2 5 punctul 3
1 2 punctul 1 nou
4 1 punctul 4
```

Asupra punctelor din fișier se fac operațiile codificate în fișierul “interogari.in” astfel: pe fiecare linie se dau două numere reprezentând coordonatele unui punct și o valoare din mulțimea {1,0} cu semnificația interogare (1)/ștergere(0).

```
1 2 1
2 4 1
1 2 0
1 3 0
4 1 1
10 20 1
20 40 0
```

Ca rezultat al operațiilor se va crea fișierul interogari.out în care în dreptul punctelor cu 1 se va scrie eticheta (sau mesajul *nu exista*), iar la final se va afișa mesajul: “punctele ramase” și apoi vor afișa punctele rămase după ștergerea punctelor cu 0 (sub forma (x,y): eticheta). Dacă un punct apare de mai multe ori în puncte.in se păstrează ultima(variante:prima/toate) etichetă asociată lui.

```
(1,2) punctul 1 nou
(2,4) nu exista
(4,1) punctul 4
(10,20) nu exista
punctele ramase
(2, 5): punctul 3
(4, 1): punctul 4
```

4. a) Scrieți o funcție care, dat numele unui fișier, determină și returnează frecvența caracterelor din fișier

b) Se consideră fișierele caractere1.in și caractere2.in. Să se afișeze pentru fiecare fișier frecvența caracterelor.

c) Să se afișeze caracterele comune celor două fișiere și frecvența cu care se repetă în ambele fișiere (se vor afișa întâi caracterele alfanumerice ordonate crescător, apoi celelalte).

5. Se citesc de la tastatură două numere naturale n și m .

a) Să se scrie în fișierul `matrice.in` o matrice cu n linii și m coloane cu elemente mai mici decât 100 generate aleator (fără a o memora). – modulul `random` (metodele `randint`, `choice`)

<https://docs.python.org/3/library/random.html>

b) Se citește din fișierul `matrice.in` matricea generată la a). Să se genereze în memorie și să se scrie în fișierul `matrice.out` transpusa matricei și matricea obținută din matricea inițială ordonând crescător elementele de pe ultima coloană prin interschimbări de linii.

6. Se dă un fișier cu cuvinte separate prin spații. Scrieți un program care să determine grupurile de cuvinte din fișier care au aceleași litere (nu neapărat cu aceeași frecvență). Numele fișierului de intrare se va citi de la tastatură, iar grupurile formate din cel puțin două cuvinte se vor scrie în fișierul text `"litere.txt"`, câte un grup pe o linie. Cuvintele din fiecare grup vor fi sortate după lungime, **iar în caz de lungimi egale, lexicografic**.

Pentru fișierul de intrare:

```
apar mare  
si amara rapa para  
par isi rama
```

fișierul de ieșire va fi

```
apar mare  
si amara rapa para  
par isi rama
```

7. Se consideră un fișier de intrare `produse.in` cu informații despre magazine, produse și cantitatea de produse din fiecare magazin, sun forma:

Magazin `cod_magazin` `nume_magazin`

`cantitate` `nume_produș`

...

`cantitate` `nume_produș`

Magazin `cod_magazin` `nume_magazin`

`cantitate` `nume_produș`

...

`cantitate` `nume_produș`

Exemplu:

```
Magazin 123 magazin1  
5 mere  
7 pere  
2 prune  
Magazin 221 magazin 2  
3.5 pere  
10 banane
```

- a) Să se memoreze datele astfel încât să răspundă la interogări de tipul: dat codul magazinului și numele unui produs, să se afișeze cantitatea de produs din magazin (să se interogheze pentru un cod și un nume date de la tastatură), dat codul magazinului, numele unui produs și o cantitate de produs care se vinde, actualizează stocul de produs din magazin dacă se poate (să se facă o astfel de actualizare pentru cod, nume produs, cantitate date de la tastatură)
- b) Să se afișeze o lista de tupluri (magazin, lista de (produse:cantitate)) ordonată după cantitatea totală de marfă din magazin, și, în caz de egalitate, după numele magazinului
- c) Să se afișeze o listă a tuturor produselor care se găsesc în magazine (folosind reuniune de mulțimi)

PROBLEME LABORATOR 4

1. a) Scrieți o funcție care, dat ca parametru numele unui fișier, determină și returnează frecvența cuvintelor din fișier; cuvintele sunt considerate separate prin caractere albe
- b) Se consideră fișierele caractere1.in si caractere2.in. Să se afișeze pentru fiecare fișier frecvența cuvintelor care apar în acesta
- c) Să se determine cuvântul care apare cel mai des în caractere1.in. Dacă sunt mai multe posibilități, se vor afișa cuvintele cele mai mici din punct de vedere lexicografic.
- d) Să se afișeze cuvintele care apar în cel puțin unul dintre cele două fișiere ordonate lexicografic și frecvența totală cu care apar (suma frecvențelor cu care apar în cele două fișiere)
- e) Pentru două documente text, F_1 și F_2 , și $\{c_1, c_2, \dots, c_n\}$ mulțimea cuvintelor care apar în cel puțin unul din cele două documente. Pentru $1 \leq i \leq n$, fie v_{i1}, v_{i2} numărul de apariții al cuvântului i în primul, respectiv în al doilea document. Distanța cosinus dintre cele două documente, notată $dcos(F_1, F_2)$, dintre F_1 și F_2 se calculează după formula:

$$dcos(F_1, F_2) = \frac{\sum_{i=1}^n v_{i1} v_{i2}}{\sqrt{\sum_{i=1}^n v_{i1}^2} \sqrt{\sum_{i=1}^n v_{i2}^2}}$$

Să se calculeze distanța cosinus dintre fișierele caractere1.in si caractere2.in

2. În fișierul "elevi.in" sunt memorate informații despre elevii unei clase; astfel, pe o linie a fișierului se dau următoarele informații despre un elev: cnp, nume (fără spații), prenume (fără spații), lista de note (maxim 5).
 - a)Memorați lista de elevi din fișier astfel încât să se poată răspundă cât mai eficient la întrebări de tipul: dat cnp elev, care sunt numele, notele.
 - b)) Scrieți o funcție care primește ca parametri un cnp și informațiile despre elevi memorate la a) și crește cu 1 prima notă a elevului. Funcția returnează nota după modificare sau None dacă cnp-ul nu există. Apelați funcția pentru un cnp citit de la tastatură

c) Scrieți o funcție care primește ca parametri un cnp și informațiile despre elevi memorate la a) și șterge informațiile despre elevul cu acest cnp. Apelați funcția pentru un cnp citit de la tastatură (dacă cnp-ul nu este în listă funcția nu va modifica nimic și nu va da eroare)

d) Folosind memorarea de la a) (nu citind din nou datele) construiți în memorie o lista cu perechi (nume, lista de note) cu elevii din fișier, ordonata după medie și, în caz de egalitate, după nume.

5234 Ion Ionica 10 8 2 9

1789 Mihail Marius 10 10 10

8923 Ionescu Maria 4 5 8 7

1789 Radulescu Matei 10 10

3. În fișierul text “test.in” se află testul unui elev de clasa a II-a la matematică, conținând 9 înmulțiri scrise pe rânduri distincte. Un calcul corect este notat cu un punct, iar unul incorect cu 0 puncte. Să se realizeze un program care să evalueze testul dat, astfel: în dreptul fiecărui calcul corect se va scrie mesajul ‘Corect’, iar în dreptul fiecărui calcul greșit se va scrie mesajul ‘Gresit’ și rezultatul corect, iar la final se va scrie nota (un punct se acordă din oficiu!). Rezultatul evaluării testului se va scrie în fișierul text “test.out”.

test.in	test.out
3*4=11	3*4=11 Gresit 12
2*10=20	2*10=20 Corect
5*5=24	5*5=24 Gresit 25
7*4=28	7*4=28 Corect
2*6=12	2*6=12 Corect
10*10=100	10*10=100 Corect
3*9=27	3*9=27 Corect
6*7=33	6*7=33 Gresit 42
0*9=1	0*9=1 Gresit 0
	Nota 6

4. Scrieți un program care să determine grupurile de cuvinte dintr-un fișier text care p-rimează între ele = au aceleași ultime p-litere (p citit de la tastatura). Numele fișierului de intrare se va citi de la tastatură, iar grupurile formate din cel puțin două cuvinte distincte se vor scrie în fișierul text “rime.txt”, câte un grup pe o linie. Cuvintele din fiecare grup vor fi sortate alfabetic.

5. În fișierul text “date.in” sunt memorate, pe linii, numele și prenumele studenților dintr-o grupă. Să se scrie un program care să genereze conturile de email ale studenților și parolele temporare, după care să le salveze în fișierul text “date.out”. Contul de email al unui student va fi de forma prenume.nume@s.unibuc.ro, iar parola temporară va fi de forma o literă mare, 3 litere mici și 4 cifre.

Pentru generarea de numere/litere aleatoare – modulul random (metodele randint,choice)

date.in

Bobocea Andrei

Marinescu Ciprian

Vasile Dragos

date.out

andrei.bobocea@s.unibuc.ro,Wadf2133

ciprian.marinescu@s.unibuc.ro,Qsdd2111

dragos.vasile@s.unibuc.ro,Bbyt7690

6. Scrieți o funcție “negative_pozitive” care primește ca parametru o lista de numere întregi și returnează două liste, prima formată din numerele strict negative din lista primită ca parametru și a doua formată din cele strict pozitive. Scrieți un program care citește de la tastatură numele unui fișier text și apoi, folosind apeluri utile ale funcției “negative_pozitive” scrie la sfârșitul fișierului text, pe două linii separate, numerele strict negative, respectiv cele strict pozitive, ordonate crescător