

## MODALITATEA DE DESFĂȘURARE A TESTULUI DE LABORATOR LA DISCIPLINA "PROGRAMAREA ALGORITMILOR"

- Testul de laborator la disciplina "Programarea algoritmilor" se va desfășura în ziua de **08.01.2022**, între orele 9<sup>30</sup> și 12<sup>00</sup>, astfel:
  - **09<sup>30</sup> – 10<sup>00</sup>**: efectuarea prezenței studenților
  - **10<sup>00</sup> – 11<sup>30</sup>**: desfășurarea testului
  - **11<sup>30</sup> – 12<sup>00</sup>**: verificarea faptului că sursele trimise de către studenți au fost salvate pe platformă
- Testul se va desfășura pe platforma MS Teams, iar pe tot parcursul desfășurării lui studenții trebuie să fie conectați pe canalul dedicat **cursului** de "Programarea algoritmilor" corespunzător seriei lor.
- În momentul efectuării prezenței, fiecare student trebuie să aibă pornită camera video în MS Teams și să prezinte buletinul sau cartea de identitate. Dacă dorește să-și protejeze datele personale, studentul poate să acopere codul numeric personal și/sau adresa!
- În timpul desfășurării testului studenții pot să închidă camera video, dar trebuie să o deschidă dacă li se solicită acest lucru de către un cadru didactic!
- Testul va conține **3 subiecte**, iar un subiect poate să aibă mai multe cerințe.
- Rezolvarea unui subiect se va realiza într-un singur fișier sursă Python (.py), indiferent de numărul de cerințe, care va fi încărcat/atașat ca răspuns pentru subiectul respectiv.
- Numele fișierului sursă Python trebuie să respecte următorul șablon: **grupa\_nume\_prenume\_subiect.py**. De exemplu, un student cu numele Popescu Ion Mihai din grupa 131 trebuie să denumească fișierul care conține rezolvarea primului subiect astfel: **131\_Popescu\_Ion\_Mihai\_1.py**.
- La începutul fiecărui fișier sursă Python se vor scrie, sub forma unor comentarii, următoarele informații: numele și prenumele studentului, grupa sa și enunțul subiectului rezolvat în fișierul sursă respectiv. Dacă un student nu reușește să rezolve deloc un anumit subiect, totuși va trebui să încarce/atașeze un fișier sursă Python cu informațiile menționate anterior!
- Toate rezolvările (fișierele sursă Python) trimise de către studenți vor fi verificate din punct de vedere al similarității folosind un software specializat, iar eventualele fraude vor fi sancționate conform Regulamentului de etică și profesionalism al FMI ([http://old.fmi.unibuc.ro/ro/pdf/2015/consiliu/Regulament\\_etica\\_FMI.pdf](http://old.fmi.unibuc.ro/ro/pdf/2015/consiliu/Regulament_etica_FMI.pdf)).

### Subiect 1

[4 p.] Fișierul text *text.in* conține pe prima linie un cuvânt *w* nevid format din litere mici ale alfabetului englez, iar pe următoarele linii un text în care cuvintele sunt despărțite prin spații și semnele de punctuație uzuale. Să se scrie în fișierul text *text.out* toate cuvintele din fișierul *text.in* care au mulțimea literelor inclusă în mulțimea literelor cuvântului *w* sau mesajul "Imposibil" dacă în fișierul de intrare nu există nici un cuvânt cu proprietatea cerută, conform modelului din exemplul de mai jos. Cuvintele vor fi scrise grupat, în funcție de mulțimile literelor. Grupele vor fi scrise în ordine lexicografică, iar în cadrul fiecărui grup cuvintele vor fi scrise în ordinea descrescătoare a lungimilor lor. Fiecare cuvânt va fi scris o singură dată și nu se va face distincție între litere mici și litere mari.

### Exemplu:

text.in	text.out
semnificare Mama s-a gandit sa isi puna seara pe masa de ceara o sama de bucate si bauturi care sunt si acre si dulci.	Literele ['a', 'c', 'e', 'r']: ceara care acre Literele ['a', 'e', 'r', 's']: seara Literele ['a', 'm']: mama Literele ['a', 'm', 's']: masa sama Literele ['a', 's']: s-a sa Literele ['i', 's']: isi si

Cuvântul "semnificare" are mulțimea literelor egală cu {'a', 'c', 'e', 'f', 'i', 'm', 'n', 'r', 's'}, iar cuvintele scrise în fișierul *text.out* sunt singurele din fișierul *text.in* care au mulțimea literelor inclusă în mulțimea literelor sale.

## Subiectul 2

**a) [0,5p]** Fișierul "**date.in**" conține  $n > 1$  linii cu următoarea structură: pe linia  $i$  se găsesc  $n$  numere naturale nenule separate prin câte un spațiu. Să se scrie o funcție **citire** care să citească datele din fișier și să returneze matricea de dimensiuni  $n \times n$  care conține numerele în ordinea din fișier.

**b) [1,5p]** Să se scrie o funcție **modifica\_matr** care primește ca parametri o matrice pătratică  $n \times n$  și un număr variabil de parametri  $x_1, x_2, \dots, x_k$  cu valori cuprinse între 0 și  $n-1$ , reprezentând indicii unor linii/coloane. Funcția va modifica matricea primită ca parametru astfel:

- adaugă o linie nouă la finalul matricei (după ultima linie existentă), în care fiecare element de pe coloana  $j$  va fi egal cu:
  - **-1**, dacă indicii  $j$  nu se află printre parametrii  $x_1, x_2, \dots, x_k$  primiți de funcție sau
  - **suma** elementelor de pe coloana  $j$  aflate strict deasupra diagonalei principale, dacă indicii  $j$  se află printre parametrii  $x_1, x_2, \dots, x_k$  primiți de funcție.
- apoi adaugă (la matricea obținută după adăugarea liniei) o coloană nouă la începutul matricei (înainte de prima coloană existentă), în care fiecare element de pe linia  $i$  va fi egal cu:
  - **-1**, dacă indicii  $i$  nu se află printre parametrii  $x_1, x_2, \dots, x_k$  primiți de funcție sau
  - **maximul** dintre elementele de pe linia  $i$  aflate pe diagonala secundară sau sub ea, dacă indicii  $i$  se află printre parametrii  $x_1, x_2, \dots, x_k$  primiți de funcție.

**c) [1p]** Să se apeleze funcția de la **b)** pentru matricea obținută la a) și indicii corespunzători următoarelor linii/coloane: prima, a doua, ultima, una respectiv două din mijlocul matricei (în funcție dacă  $n$  este impar respectiv par). Matricea obținută să se afișeze pe ecran, fără paranteze și virgule, iar elementele de pe fiecare coloană să fie aliniate la dreapta ținând cont că numerele pot avea maxim 4 caractere (inclusiv semnul '-').

date.in	Afișare pe ecran
1 2 3 4 5 6 7	7 1 2 3 4 5 6 7
2 8 2 3 1 5 4	5 2 8 2 3 1 5 4
3 1 4 2 6 3 3	-1 3 1 4 2 6 3 3
4 7 1 5 8 3 6	8 4 7 1 5 8 3 6
5 3 7 8 2 9 2	-1 5 3 7 8 2 9 2
6 9 1 7 4 2 8	-1 6 9 1 7 4 2 8
7 5 2 6 8 4 1	8 7 5 2 6 8 4 1
	-1 0 2 -1 9 -1 -1 30

Explicații: După modificări, se va obține matricea:

7	1	2	3	4	5	6	7
5	2	8	2	3	1	5	4
-1	3	1	4	2	6	3	3
8	4	7	1	5	8	3	6
-1	5	3	7	8	2	9	2
-1	6	9	1	7	4	2	8
8	7	5	2	6	8	4	1
-1	0	2	-1	9	-1	-1	30

### Subiect 3

Se consideră fișierul text **catalog.in** cu următoarea structură:

- pe prima linie apare numărul **n** reprezentând *numărul* de elevi dintr-o clasă a unui liceu
- pe următoarele **linii** avem informații despre cei **n** elevi, respectiv pentru fiecare elev informațiile sunt structurate astfel:
  - linie de forma *<șir de caractere> <m>*, unde *șirul de caractere* este numele elevului (acesta este unic), iar *m* este un număr natural reprezentând numărul de materii
  - urmată de **m** linii care conțin notele elevului (numere naturale) la **m** materii, fiecare având următoarea structură:

*<nume\_materie>,<nota\_1>,<nota\_2>,...,<nota\_k>*

**Observație:** Orice elev are la fiecare materie cel puțin o notă, iar denumirile materiilor nu conțin caracterul ',' (virgula).

#### Exemplu de fișier de intrare:

```
5
Ana Maria Pop 3
Matematica,10,9,9,10,10
Limba romana,8,9,9,8
Fizica,10,9,7,10,10
Mihai Popescu 3
Matematica,9,7,10,10
Limba romana,8,3,5,10
Fizica,10,10
Andrei Mincu 2
Matematica,10,9,2
Fizica,3,7,9
Ioana Matei 3
Fizica,10,10
Matematica,10,10,10,9
Limba romana,9,9,10,10
Alin Enache 3
Limba romana,10,10,10
Matematica,10,10,10,10
Fizica,10
```

#### Cerințe:

- [2 p.]** Scrieți o funcție care citește datele din fișierul **catalog.in** și returnează o structură de date cu informațiile din fișier. Folosiți o structură de date convenabilă pentru a rezolva eficient subpunctele următoare.
- [1 p.]** Scrieți o funcție **detalii\_elev** care primește ca parametri structura în care s-au memorat datele la cerința a) și un șir de caractere reprezentând **numele** unui elev și returnează mediile la toate materiile elevului cu numele primit ca parametru, memorate sub formă de listă de tupluri de tipul (*nume\_materie, medie*). Dacă un elev are *o singură notă* la o materie **sau** media este *mai mică strict decât 5*, acesta va avea media egală cu 0 și va rămâne **corigent**. Să se citească de la tastatură numele unui elev și să se afișeze pe ecran mediile acestuia (rotunjite cu două zecimale) la fiecare materie (sortate lexicografic) folosind această funcție.

**Exemplu:**

Intrare **tastatură:**  
Ana Maria Pop

**Afișare pe ecran:**  
Fizica 9.20  
Limba romana 8.50  
Matematica 9.60

- c) [1 p.] Scrieți o funcție **clasament** care primește structura de date în care s-au memorat datele la cerința a) și un număr variabil de parametri de tip șir de caractere reprezentând nume de elevi. Funcția returnează o listă de tupluri de tipul (*nume\_elev*, *medie\_generala*) cu mediile generale ale elevilor ale căror nume au fost primite ca parametru ordonată descrescător după medii. *Media generală* a unui elev este egală cu media aritmetică a mediilor de la fiecare materie, dacă acesta nu este corigent, altfel media este 0.

**Exemplu:** Dacă se apelează funcția pentru elevii *Alin Enache* și *Ioana Matei* se va returna lista [(Ioana Matei,9.75), (Alin Enache,0)], deoarece *Alin Enache* are o singură notă la fizică, deci este corigent.