

## TEST DE LABORATOR LA DISCIPLINA “PROGRAMAREA ALGORITMILOR” VARIANTA 2

### Subiectul 1 – 4 p.

**a) [0,5 p.]** Scrieți o funcție *citire\_siruri* cu un parametru reprezentând numele unui fișier text care conține, pe mai multe linii, cuvinte scrise doar cu litere mici și despărțite între ele prin spații și returnează o listă de liste (numite subliste), elementele unei subliste fiind cuvintele de pe o linie din fișier.

**b) [2 p.]** Să se scrie o funcție *prelucrare\_siruri* care primește ca parametru o listă de liste  $L$  de tipul celei returnate la punctul a) și un număr natural  $n \geq 1$  și modifică lista  $L$  astfel:

- la sfârșitul fiecărei subliste din lista  $L$  se va adăuga cuvântul format prin alipirea ultimei litere din fiecare cuvânt din sublista respectivă, apoi
- din fiecare sublistă se vor elimina toate cuvintele care nu conțin cel puțin  $n$  vocale.

**c) [0,5 p.]** Se dă fișierul *cuvinte.in* având structura precizată la punctul a). Să se apeleze funcția *prelucrare\_siruri* pentru lista de liste  $L$  obținută în urma apelului funcției *citire\_siruri* pentru fișierul text *cuvinte.in* și parametrul  $n$  egal cu 3. Lista de liste astfel obținută să se afișeze pe ecran, fără paranteze și virgule, fiecare sublistă pe câte o linie, iar elementele din fiecare sublistă să fie separate între ele prin câte un spațiu.

**d) [1 p.]** Fie  $L$  lista de liste obținută în urma apelării funcției *citire\_siruri* pentru fișierul text *cuvinte.in*. Să se citească de la tastatură un cuvânt  $w$  și apoi să se scrie în fișierul text *cuvinte.out* cuvintele din lista  $L$  care-l conțin pe  $w$  sau mesajul “*Imposibil!*” dacă nu există niciun cuvânt cu proprietatea cerută. Cuvintele vor fi scrise pe prima linie din fișier, despărțite între ele prin câte un spațiu, fără duplicate și în ordine alfabetică.

### Exemplu:

cuvinte.in	Punctul c) ecran	Punctul d) fișierul cuvinte.out pentru $w = \text{"er"}$
ana are mere si pere frumoase dar are mai multe mere rosii mici decat pere galbene si mari	frumoase aeeiee reie rosii  galbene teeii	mere pere

## Subiectul 2 – 5 p.

Cei șapte pitici sunt înscriși la Jocurile Pădurii, un concurs în curs de desfășurare, care conține două probe: sărituri pe trambulină și aruncarea greutății. Fiecare concurent a participat deja la una sau două dintre probe, iar rezultatele obținute până acum sunt scrise în fișierul **concurs.in**, având următoarea structură:

- dacă rândul conține un singur cuvânt (șir de caractere fără spații), atunci acesta reprezintă **numele** unui concurent
- sub el vor urma unul sau două rânduri de forma **nume\_proba valori\_numerice**, unde **nume\_proba** este un șir de caractere care nu conține spații, iar **valori\_numerice** sunt mai multe numere separate prin câte un spațiu; la proba de trambulină sunt numere naturale reprezentând înălțimile săriturilor (în centimetri), iar la proba cu greutăți sunt numere reale reprezentând greutatea obiectelor aruncate (în kilograme).

Un exemplu de astfel de fișier este:

### concurs.in

```
Hapciu
trambulina 20 15 25 30 25
greutati 0.25 0.75 1.5 1.25
Rusinosul
greutati 1.0 0.5 1.25 1.5
Mutulica
greutati 0.75 0.25 1.0 0.25 1.25
trambulina 35 20 20 15 30
Inteleptul
trambulina 20 30 25 15 15 20
Voiosul
greutati 1.25 1.5 1.75 1.0
trambulina 25 30 35 40 20 30
Somnorosul
trambulina 15 25 20
Morocanosul
greutati 0.75 1.0 0.25 1.5 1.25
```

**a) [2,5 p.]** Să se memoreze datele din fișier într-o singură structură de date astfel încât să se răspundă cât mai eficient la cerințele de la punctele următoare.

**b) [1,5 p.]** Scrieți o funcție **rezultate** care primește următorii parametri (în această ordine):

- structura în care s-au memorat datele la cerința a),
- un număr variabil de șiruri distincte de caractere reprezentând nume de probe existente în concurs,
- un număr natural  $N$ .

Funcția returnează o listă de tuple cu elemente de forma (*nume\_proba*, *nume\_concurent*, *medie\_nr\_ramase*, *lista\_nr\_ramase*), pentru acei concurenți care au cel puțin  $N$  valori numerice obținute la proba *nume\_proba* primită ca parametru, unde:

- *lista\_nr\_ramase* este o listă ordonată descrescător care conține toate valorile numerice obținute de *nume\_concurent* la *nume\_proba*, cu excepția unei celei mai mici și a unei celei mai mari valori care nu participă la calcularea rezultatelor, iar
- *medie\_nr\_ramase* este media aritmetică a elementelor din *lista\_nr\_ramase*, calculată cu maxim 2 zecimale.

Lista returnată va fi ordonată crescător după *nume\_proba*, apoi, în caz de egalitate, descrescător după *medie\_nr\_ramase* și apoi crescător după *nume\_concurent*.

Să se apeleze funcția *rezultate* pentru probele 'trambulina' și 'greutati' și numărul  $N = 5$ , apoi să se afișeze pe ecran lista returnată.

**Explicații:** pentru datele din fișier lista returnată va fi

```
[('greutati', 'Morocanosul', 1.0, [1.25, 1.0, 0.75]),  
( 'greutati', 'Mutulica', 0.67, [1.0, 0.75, 0.25]),  
( 'trambulina', 'Voiosul', 30.0, [35, 30, 30, 25]),  
( 'trambulina', 'Hapciu', 23.33, [25, 25, 20]),  
( 'trambulina', 'Mutulica', 23.33, [30, 20, 20]),  
( 'trambulina', 'Inteleptul', 20.0, [25, 20, 20, 15])]
```

c) [1 p.] Scrieți o funcție **adaugare** care are următorii parametri (în această ordine):

- structura în care s-au memorat datele la cerința a),
- un șir de caractere *nume\_proba* reprezentând numele unei probe,
- un șir de caractere *nume\_concurent* reprezentând numele unui concurent,
- o listă de valori numerice (numere naturale sau reale) *lista\_nr* reprezentând rezultatele obținute de *nume\_concurent* la *nume\_proba*.

Funcția va adăuga în structură aceste rezultate (dacă *nume\_concurent* a mai participat la *nume\_proba*, atunci valorile din *lista\_nr* se vor adăuga în lista valorilor existente) și va returna numărul total de rezultate obținute de *nume\_concurent* la *nume\_proba*. Dacă *nume\_proba* sau *nume\_concurent* nu există deja în structură, atunci se va returna un mesaj de forma "Numele probei sau al concurentului nu exista!", iar structura va rămâne nemodificată.

Se citesc de la tastatură un nume de probă *p*, un nume de concurent *c* și mai multe valori numerice (numere naturale sau reale) introduse pe aceeași linie cu spațiu între ele. Să se apeleze funcția *adaugare* pentru a completa structura cu rezultatele obținute de concurentul *c* la proba *p* și să se afișeze rezultatul funcției (numărul de valori numerice sau mesajul). După apelul funcției, să se afișeze și structura în care s-au memorat datele.

## MODALITATEA DE DESFĂȘURARE A TESTULUI DE LABORATOR

- Testul de laborator la disciplina "Programarea algoritmilor" se va desfășura în ziua de **15.01.2023**, în două runde, între orele 9<sup>00</sup> și 11<sup>00</sup>, respectiv 11<sup>30</sup> și 13<sup>30</sup>, astfel:
  - **Prima rundă**
    - 09<sup>00</sup> – 09<sup>15</sup>: efectuarea prezenței studenților
    - 09<sup>15</sup> – 10<sup>45</sup>: desfășurarea testului
    - 10<sup>45</sup> – 11<sup>00</sup>: trimiterea surselor folosind un formular Google dedicat
  - **A doua rundă**
    - 11<sup>30</sup> – 11<sup>45</sup>: efectuarea prezenței studenților
    - 11<sup>45</sup> – 13<sup>15</sup>: desfășurarea testului
    - 13<sup>15</sup> – 13<sup>30</sup>: trimiterea surselor folosind un formular Google dedicat
- Testul de laborator se va desfășura în laboratoarele Facultății de Matematică și Informatică, folosind calculatoarele din ele.
- Calculatoarele din laboratoare vor conține documentația oficială a limbajului Python, în format offline.
- Pe parcursul testului este interzisă utilizarea Internet-ului sau a oricărei alte forme de comunicare/informare, cu excepția documentației offline.
- În momentul efectuării prezenței, fiecare student trebuie să prezinte buletinul sau cartea de identitate.
- Testul va conține **două 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, numele complet al studentului și grupa sa. Dacă un student nu reușește să rezolve deloc un anumit subiect, totuși el 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)).