

TEST DE LABORATOR LA DISCIPLINA “PROGRAMAREA ALGORITMILOR”
Sesiunea de examene ianuarie 2025 – VARIANTA 3

Subiectul 1 – 2 p

Fișierul *matrice.in* conține elementele unei matrice (pe fiecare linie din fișier sunt elementele unei linii din matrice separate cu un spațiu), numere naturale nenule. Să se memoreze matricea din fișier într-o variabilă, apoi să se șteargă din matrice elementele care sunt palindroame. Să se adauge un număr minim de 0-uri la finalul liniilor mai scurte până când toate liniile matricei au aceeași lungime. Să se scrie matricea obținută în fișierul *matrice.out* (pe fiecare linie din fișier se vor scrie elementele unei linii din matrice separate prin câte un spațiu).

Exemplu:

matrice.in	matrice.out
11 343 57 222	57 0 0
48 100 521 303	48 100 521
77 82 949 25	82 25 0
1 88 262 505	0 0 0

Subiectul 2 – 2 p

Un șir de litere se poate restrânge folosind paranteze rotunde astfel: șirul ababab poate fi scris simplificat $(ab)^3$, iar axyzxyzw va fi scris $a(xyz)^2w$. Pe fiecare linie a fișierului text *retrans.in* se află câte un șir restrâns (în care nu sunt prezente paranteze una în alta). Scrieți un program care determină șirul inițial pentru fiecare astfel de șir și salvează șirurile obținute în fișierul *retrans.out*, separate prin virgulă.

Exemplu:

retrans.in	retrans.out
$(a)^3bc(doi)^2$	
$m(a)^{10}te$	
casa	aa a bc do id oi oi,maaaaaaaaaate,casa

Subiectul 3 – 5 p

Fișierul *depozite.in* conține informații despre depozitele bancare ale unei persoane. Pentru fiecare bancă la care are depozit informațiile încep cu o linie de forma

Banka: nume_banca , Moneda: moneda

~~01234567~~ unde **nume_banca** este numele băncii la care sunt deschise depozitele (și nu conține virgulă), **moneda** este moneda în care sunt făcute depozitele, urmată de mai multe linii de forma:

suma_dobanda_perioada

SERIILE 13, 14 și 15 – INFORMATICĂ
ANUL UNIVERSITAR 2024-2025

reprezentând informațiile despre fiecare depozit în acea monedă de la acea bancă, unde:

- **suma**: suma depozitată, exprimată ca număr întreg,
- **dobanda**: dobânda depozitului, exprimată ca număr real (în procente),
- **perioada**: perioada depozitului, exprimată în luni (număr întreg).

Un exemplu de astfel de fișier este:

depozite.in

Banca: Banca Transilvania , Moneda: LEI
10000 4.5 12
5000 4.0 6
Banca: Banca Transilvania , Moneda: EURO
2000 2.0 12
Banca: ING Bank , Moneda: DOLAR
1500 3.5 24
Banca: ING Bank , Moneda: LEI
7000 4.2 18
3000 4.0 6
Banca: Raiffeisen Bank , Moneda: LIRA
1000 2.8 12
Banca: Raiffeisen Bank , Moneda: EURO
3000 2.5 36
Banca: BRD , Moneda: LEI
12000 4.3 24
8000 4.1 12
Banca: Alpha Bank , Moneda: EURO
5000 2.3 18
1000 2.0 6

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 p.] Scrieți o funcție **actualizeaza_depozite** care primește următorii parametri (în această ordine):

- structura în care s-au memorat datele la cerința a),
- un număr variabil de șiruri de caractere reprezentând nume de bănci,
- un șir de caractere **m** reprezentând o monedă,
- un număr real **p** reprezentând procentul cu care se va mări dobânda,
- un număr întreg **x** care reprezintă perioada minimă (în luni) a depozitelor ce vor fi modificate.

Funcția va parcurge depozitele bancare din băncile primite ca parametru și va mări dobânda cu **p** doar pentru depozitele din moneda **m** specificată, în cazul în care perioada acestora este mai mare sau egală cu **x** luni. După apelul funcției, se va afișa structura actualizată a depozitelor bancare. Să se apeleze funcția pentru băncile Banca Transilvania și Alpha Bank, moneda LEI, procentul 2 și numărul de luni 12.

Explicații: dacă lista de bănci este: Banca Transilvania și Alpha Bank, moneda este LEI procentul este 2 și numărul de luni este 12, în structura de date se va modifica doar depozitul corespunzător liniei 10000 4.5 12, noua dobândă fiind 6.5 (doar acesta are o durată de minim x=12 luni)

c) [1,5 p.] Scrieți o funcție **centralizator** care primește ca parametru structura de date creată la cerința a) și returnează o listă de tupluri de forma:
(moneda, suma_initiala_moneda, suma_finala_moneda)
unde:

- **moneda** este moneda depozitelor (de exemplu: "LEI", "EURO", "DOLAR", "LIRA"),
- **suma_initiala_moneda** este suma totală a depozitelor în moneda respectivă înainte de aplicarea dobânzilor (rotunjită la două zecimale),
- **suma_finala_moneda** este suma totală a depozitelor în moneda respectivă după aplicarea dobânzilor (rotunjită la două zecimale).

Lista returnată trebuie să fie sortată descrescător după **câștigul** obținut, calculat ca diferență între **suma_finala_moneda** și **suma_initiala_moneda** iar în caz de egalitate, lexicografic (alfabetic) după numele **monedei**. Apelați funcția și afișați informațiile pe ecran formatare în următorul mod:

Moneda ...: suma initiala = ... suma finala = ...

Pentru datele inițiale se va afișa

Moneda LEI: suma initiala = 45000.00, suma finala = 46908.00

Moneda EURO: suma initiala = 11000.00, suma finala = 11250.00

Moneda DOLAR: suma initiala = 1500.00, suma finala = 1552.50

Moneda LIRA: suma initiala = 1000.00, suma finala = 1028.00