

# PROGRAMAREA CALCULATOARELOR

## - TEME DE CAS -

1. S se implementeze un program care s determine maximul de pe diagonalele unei matrice. S se schimbe apoi locul lui  $a_{1,1}$  cu locul maximului, celelalte elemente r mând neschimbate. Se vor avea în vedere atât matrice de ordin par cât i impar.

Acela i algoritm se va aplica i asupra unei matrice con inând *caractere*.

2. S se scrie un program care cite te de la dispozitivul standard de intrare o linie de text terminat cu un sfâr it de linie (' \0'). Delimita i cuvintele din cadrul acestei linii de text i afi a i-le pe linii separate.

Ca *separatori* se consider caracterele: spa iu ( ), tab, virgul (,), punct i virgul (;), punct (.), dou puncte (:), semnul întreb rii (?) i semnul exclam rii (!). Se vor folosi tehnicile specifice *aloc rii dinamice*.

3. S se scrie un program care cite te de la dispozitivul standard de intrare dou propozi ii, fiecare propozi ie terminat cu un caracter sfâr it de linie (' \0'). Afia i literele comune celor dou propozi ii. Se vor folosi tehnicile de *alocare dinamic*.

4. S se scrie un program care determin i afi eaz la dispozitivul standard de ie ire procentele de apari ie a literelor din cadrul unui fi ier text. Caracterele diferite de litere vor fi ignorate. Literele mari (majuscule) vor fi tratate la fel cu cele mici (minuscule). Afia area procentelor sa va face în ordinea descresc toare a valorilor lor. Se vor folosi tehnicile specifice *aloc rii dinamice*.

*Observa ie:* în locul fi ierului se poate folosi i un ir de caractere de lungime impus . Acesta se va citi de la dispozitivul de intrare (tastatur ).

5. S se scrie un program care determin to i termenii irului Fibonacci care nu dep esc un num r natural nenul dat. Pentru generarea numerelor Fibonacci se va folosi o *func ie recursiv*.

6. Determina i posibilit ile de reprezentare a unei sume de  $N$  lei folosind monede i bancnote cu diviziunile: 1, 5, 10, 20, 50, 100 lei. Afia i la dispozitivul standard de ie ire aceste posibilit i sub form de succesiuni de valori, fiecare solu ie a problemei *pe câte o linie distinct*.

7. Folosind func ii (programare func ional ) scrie i un program în care s descompune i un num r natural dat, ca sum de puteri distincte ale lui 2.

*Exemplu:*  $170 = 2^7 + 2^5 + 2^3 + 2^1$

8.(\* ) S se implementeze un program de calcul a unui determinant de ordinul  $n$ .

9.(\* ) S se implementeze un program de calcul a rangului unei matrice.

10. S se implementeze un program de calcul a produsului a dou matrice.

11. Dându-se urm torul sistem linear p tratic, calcula i solu iile folosind metoda lui CRAMER:

$$\begin{cases} 2x_1 - x_2 + 0.5x_3 = 1.5 \\ -1.7x_1 + 4x_2 - 11x_3 = -8.7 \\ 3x_1 + 0.9x_2 + 8x_3 = 11.9 \end{cases}$$

12. Se consider un vector de  $n$  numere reale. Determina i elementul maxim i minim ale vectorului, precum i toate rangurile pe care apar aceste elemente în vector. Dac vectorul con ine

elemente în progresie geometrică, concepe și rutina care să calculeze suma progresiei respective. Pentru vector se vor utiliza tehnicile specifice alocării dinamice.

13. Fie un șir de  $n$  caractere (modelat ca un *vector de caractere*). Scrieți programul care găsește cel mai lung sub-șir ordonat crescător format din elemente de ranguri succesive ale vectorului inițial (preluat la intrare). Nu se va folosi funcția de bibliotecă asociată acestei operații. În schimb, pentru toate celelalte operații pe șiruri pe care le implică soluția imaginată de către programator se pot folosi funcțiile dedicate șirurilor (modelate în C drept vectori de caractere) întâlnite în fișierul `antet_string.h`.

14. Se consideră un vector de  $n$  numere reale sau întregi. Să se scrie un program care determină :

- elementul maxim din sub-șirul format din termenii de rang *impar* ai vectorului inițial.
- elementul minim din sub-șirul format din termenii de rang *par* ai vectorului inițial precum
- rangurile comune pe care apar în șirul inițial elementele determinate anterior.

Se vor folosi cât mai puține variabile.

15. Considerându-se un vector de  $n$  numere reale, să se conceapă programul care să elimine din acest vector elementele care se repetă, folosindu-se cât mai puține variabile de lucru. Vectorii sunt considerați variabile.

16. Fie un șir (modelat ca un *vector de caractere*) citit în ordine descrescătoare. Se citește o variabilă  $v$  (o literă). Se cere un program care să insereze variabila  $v$  în poziția corespunzătoare în șir, dacă nu se află deja acolo. Dacă  $v$  este prezent în șir să se afișeze toate pozițiile pe care aceasta apare.

17. Afișați *vectorul intersecție* a două mulțimi de valori,  $V_1$  și  $V_2$ . Aplicați algoritmul asupra următoarelor mulțimi:

$$V_1 = \{-1.3, 0, 1, 2.2, 4, 6\},$$

$$V_2 = \{-1.3, 0.9, 1.1, 1.2, 4, 8\}$$

18. Cu ajutorul unui program de calcul a ecuației unei drepte din planul  $xOy$  sub formă de determinant, aflați ecuațiile dreptelor ce trec prin:

a)  $(0, 4)$  și  $(2, 6)$

b)  $(-1, 2)$  și  $(4.5, 5)$

c) panta  $m=3$  și punctul  $(-1, 1)$

d) panta  $m=1.5$  și punctul  $(0.3, 3)$ .

19. Pentru o matrice de  $m$  linii și  $n$  coloane (dreptunghiulară) ce conține caractere să se afișeze coloanele ce reprezintă șiruri ordonate crescător și liniile care reprezintă șiruri ordonate descrescător. Același algoritm și pentru valori de altă natură decât caracterele. *Indicați*: utilizați funcțiile de bibliotecă pe șiruri (prototipurile lor se găsesc în fișierul `antet_string.h`).

20. Aduceți elementul  $a_{n,n}$  al unei matrice pătratice de ordin  $n$ , în poziția  $a_{1,1}$ , pe un traseu în formă de zig-zag. Deplasarea dintr-o poziție în alta se face pe lungime de maxim două poziții stânga sau dreapta față de poziția anterioară.

*Indicați*: deplasarea nu trebuie să lase urme. Singurele valori schimbate în matricea finală sunt cele două indicate în textul problemei.

21. Pentru o matrice pătratică  $A_{n,n}$ , pentru toate sub-matricile pătratice formate din liniile și coloanele  $i$ , cu  $i=1, \dots, n$ , calculați elementele maxime în valoare absolută. Acestea vor fi aduse pe pozițiile  $a_{n,n}$ . Determinați apoi dacă matricea este *dominant-diagonală*: pe fiecare linie a matricei, elementul de pe diagonala principală în valoare absolută  $|a_{i,i}|$  este strict mai mare decât suma modulelor vecinilor săi de pe linia  $i$ .

22. Să se calculeze media aritmetică și produsul elementelor cuprinse în zonele triunghiulare superioare și inferioare, delimitate de diagonalele principală și secundară ale unei matrice cu tip de date la alegere. Se vor afișa valorile cuprinse în zonele indicate. Toate valorile de lucru vor fi preluate de la utilizator.

23. Dacă rândurile unei matrice  $A_{n,n}$  reprezintă iruri de caractere, ordonate crescător pentru indicii impari, și descrescător pentru cei pari. Aduce irurile ordonate descrescător pe primele rânduri ale matricei, iar pe cele ordonate crescător pe ultimele rânduri.

24. Să se creeze o structură de tip Persoană, cu cinci câmpuri la alegere. Declarați patru variabile de acest tip și inițializați-le. Afișați câmpurile variabilelor structură pentru care anul de naștere este mai mare decât 1980. Utilizați operații/operatori de alocare dinamică pentru toate variabilele cerute în problemă.

*Operațional:* Folosiți aceste variabile în construcția unei liste de persoane, urmând să utilizați operații specifice acestei structuri derivate de date (list) pentru a îndeplini cerințele problemei.

25.(\*). Dați o soluție *problemei rucsacului*. Folosind operațiile tipice fișierelor, preluați datele de intrare și salvați soluția în câte un fișier separat, cu denumiri sugestive, la alegere. În prealabil, dați o scurtă descriere a algoritmului pe care îl implică problema.

26.(\*). Dați o soluție *problemei comis-voiajorului*. Folosind operațiile tipice fișierelor, preluați datele de intrare și salvați soluția în câte un fișier separat, cu denumiri sugestive, la alegere. În prealabil, dați o scurtă descriere a algoritmului pe care îl implică problema.

27.(\*). Dați o soluție *problemei damelor*. Folosind operațiile tipice fișierelor, preluați datele de intrare și salvați soluția în câte un fișier separat, cu denumiri sugestive, la alegere. În prealabil, dați o scurtă descriere a algoritmului pe care îl implică problema.

28.(\*). Dați o soluție *problemei drumului minim* într-un graf. Folosind operațiile tipice fișierelor, preluați datele de intrare și salvați soluția în câte un fișier separat, cu denumiri sugestive, la alegere. În prealabil, dați o scurtă descriere a algoritmului pe care îl implică problema.