



## PROGRAMARE ALGORITMI C++ LABORATOR 5

### Obiective

1. functii: *definitie, prototip, parametrii formali*
2. definitia recursiei: *directa, indirecta*
3. exprimarea recursiei: *cazuri de baza, cazuri generale*
4. manipularea sirurilor folosind recursia
5. metoda *divide et impera*
6. declararea structurilor
7. accesarea membrilor unei structuri: *operatorii . si ->*
8. structuri, functii si asignari
9. initializarea structurilor
10. *typedef*
11. structuri recursive: *liste liniar inlantuite*
12. operatii pentru liste: *creare, inserare, stergere, cautare, numarare*
13. functii de intrare/iesire: *sprintf(), sscanf(), fprintf(), fscanf()*
14. accesarea fisierelor: *fstream*
15. cronometrarea executiei unui program C++

➤ **Subprograme (recursie). Programe complexe. Siruri si structuri. Liste inlantuite. Fisiere si instrumente soft**

### ➤ CERINTE

1. 10 probleme din primele 18 pentru nota 7; 14 probleme pentru 8; 18 probleme pentru 9;
2. suplimentar inca 5 din problemele 19, ..., 28 pentru nota 10.

### ➤ EXERCITII

1. Scrieti propria voastra functie *quicksort()* care sa fie echivalenta cu *qsort()* pus la dispozitie de sistemul C.
2. Definiti o functie pentru XOR(), numita *sau exclusiv*. Un apel XOR(a,b)=true <=> a este true si b false, sau a false si b true. Scrieti si o functie XOR(a,b,c) si una XOR(a,b,c,d).
3. Scrieti o functie C++ recursiva care calculeaza media aritmetica a unui sir de numere reale.
4. Scrieti o functie C++ recursiva care calculeaza n!, unde n este un numar natural.
5. Sa se calculeze suma cifrelor unui numar cu n cifre (numarul se reprezinta ca vector).



6. (Operatii cu liste) a) Concatenarea a doua liste; b) Determinarea unei subliste ce contine primele  $k$  elemente dintr-o lista, cu eliberarea zonelor de memorie ale restului elementelor.
7. Sa se sorteze  $n$  numere folosind liste liniar inlantuite si metoda interclasarii.
8. Scrieti un program C++ in care sa descrieti urmatoarele operatii pentru arbori binari: creare, numarare, cautare, stergere, inserare, parcurgere (preordine, inordine, postordine).
9. Scrieti o functie C++ de inmultire a polinoamelor, care se bazeaza pe urmatoarea metoda:  

$$(a_0 + a_1 * X + .. a_n * X^n) * (b_0 + b_1 * X + .. b_n * X^n) =$$

$$a_0 * (b_0 + b_1 * X + .. b_n * X^n) + ... + a_n * X^n * (b_0 + b_1 * X + .. b_n * X^n).$$
10. Scrieti si o functie C++ care calculeaza catul si restul impartirii a doua polinoame.
11. *Oglinditul* unui cuvânt  $w = a_1 \dots a_n$  este un cuvânt  $w' = a_n \dots a_1$ . Scrieti un program care pentru un sir dat  $w$ :
  - o determina oglinditul sau  $w'$
  - o decide daca  $w$  este de forma  $uu'$
12. Scrieti un program care, avand la intrare doua siruri  $x=x_1 \dots x_m$  si  $y=y_1 \dots y_m$ , construiesc sirurile:
  - o  $x_1 \dots x_m y_1 \dots y_m$ .
  - o  $m(x,y)=x_1 y_1 \dots x_m y_m$ .
  - o  $m'(x,y)=x_m y_m \dots x_1 y_1$ .
13. Program de "citire" a numerelor intregi cu cel mult 8 cifre (pentru un  $n$  dat, construiti sirul care il exprima in vorbirea curenta).
14. Proiectati structuri de date pentru reprezentarea unui punct, a unui triunghi, dreptunghi, cerc.
  - o Scrieti proceduri de citire si scriere a unui punct, triunghi etc.
  - o Scrieti o procedura care determina, pentru un triunghi dat, numarul punctelor de coordonate intregi aflate in interiorul sau.
  - o Scrieti o procedura similara pentru cerc.
15. Proiectati o structura de date pentru reprezentarea datei calendaristice si scrieti subprograme care:
  - o verifica daca valoarea unei variabile din structura este o data valida.
  - o calculeaza data calendaristica care urmeaza unei anumite date.
  - o calculeaza data calendaristica care precede o anumita data.
16. Scrieti un program C++ pentru simularea unui joc de carti. Sa se faca o amestecare si o impartire a cartilor la cei  $n$  jucatori.
17. Folosind subprograme, sa se scrie trei functii C++ care sa calculeze suma a doi vectori, produsul scalar a doi vectori si inmultirea matricelor.
18. Scrieti un program C++ care implementeaza strategiile "bubble sort", respectiv "quicksort", si folosind functii "de timp" comparati timpii de executie ale celor doua metode.
19. (\*) Fie multimea  $S=\{0, 1, \dots, m-1\}$ . O submultime  $M$  a sa poate fi reprezentata printr-un *vector caracteristic*  $c$ , cu elemente 0 si 1, unde  $c[i]=1$  daca elementul  $i$  este in  $M$  si  $c[i]=0$  daca  $i$  nu este in  $M$ .
  - o Sa se scrie procedura care calculeaza succesorul unui vector caracteristic (peste multimea vectorilor caracteristici se considera ordinea lexicografica)
  - o Scrieti un program care afiseaza toate submultimile lui  $S$ .



- Scrieti un program care afiseaza toate submultimile lui  $S$  cu  $k$  elemente ( $k$  dat).
20. (\*) Scrieti un program care genereaza toate permutarile multimii  $S_n = \{1, 2, \dots, n\}$ .
  21. (\*) (Mutarea calului) Data o tabla de sah ( $8 \times 8$ ), sa se scrie o functie C++ recursiva care descrie mutarile calului astfel incat orice pozitie sa fie parcursa o singura data.
  22. (\*) Sa se oglindeasca o lista liniara inlantuita cu numar constant de variabile suplimentare fara a folosi recursie.
  23. (\*) Scrieti un program C++ in care sa descrieti aceleasi operatii de mai sus, dar pentru arbori generali (idee: folositi legaturi de tip fiu-frate).
  24. (\*) Scrieti un program C++ pentru reuniunea a doua multimii folosind operatori pe biti. Multimile  $A$  si  $B$  sunt incluse in multimii de forma  $\{0, \dots, n_1\}$  si  $\{0, \dots, n_2\}$ .
  25. (\*) Rezolvati problema precedenta in cazul vectorilor, respectiv matricelor *rare* (adica cu multe elemente 0).
  26. (\*) (Problema lui Joseph)  
Fie  $n$  persoane aranjate in cerc. Incepand cu o pozitie  $i$ , se numara persoanele din cerc si se *executa* fiecare a  $m$ -a persoana. Cercul se inchide la loc dupa fiecare executie. "Jocul" se incheie cand ramane o singura persoana.  
Scrieti un program care tipareste numele persoanelor in ordinea in care au fost executate.  
Pp. ca doriti sa salvati viata unei persoane, determinati pozitia  $i$  de la care incepe numaratoarea, astfel incat persoana respectiva sa fie ultima ramasa (deci, salvata).
  27. (\*) *Polinoamele rare* cu coeficienti intregi sunt polinoame de grade mari cu multi coeficienti egali cu 0 si pot fi reprezentate printr-o structura de date definita astfel:

```
typedef struct
{
    int Coef;
    unsigned int Exponent;
} TMonom;
typedef TMonom TPolinom[50];
```

Scrieti proceduri de scriere, citire, adunare, inmultire a polinoamelor rare reprezentate astfel.

28. (\*) Scrieti un program care, avand la intrare doua siruri de caractere, determina cel mai lung subsir comun si pozitiile la care acesta incepe in fiecare dintre cele doua siruri.