

Programare procedurală

– Laborator 10 –

1. Scrieți un program care citește de la tastatură o matrice pătratică de dimensiune $n \geq 2$ și construiește o altă matrice formată din elementele matricei inițiale, mai puțin cele aflate pe diagonala principală. Ambele matrice vor fi alocate dinamic!
2. Scrieți un program care citește de la tastatură un număr natural nenul $n \geq 2$ și construiește un tablou bidimensional triunghiular alocat dinamic, după următoarele reguli:
 - prima coloană conține numerele de la 1 la n , în ordine descrescătoare;
 - ultima linie conține numerele de la 1 la n , în ordine crescătoare;
 - orice alt element este egal cu suma vecinilor săi de la vest și sud.

Exemplu: Pentru $n = 4$ programul trebuie să afișeze următorul tablou:

4			
3	7		
2	4	7	
1	2	3	4

3. Fișierul text *siruri.txt* are următoarea structură: pe prima linie un număr natural nenul n_1 , pe a doua linie n_1 numere întregi despărțite între ele prin câte un spațiu, pe a treia linie un număr natural nenul n_2 , pe a patra linie n_2 numere întregi despărțite între ele prin câte un spațiu ș.a.m.d.
 - a) Folosind funcțiile pentru alocarea/realocarea dinamică a memoriei, creați o structură de date bidimensională care să permită memorarea tuturor șirurilor de numere întregi din fișierul text dat. Pe fiecare linie, prima valoare va reprezenta numărul de valori de pe linia respectivă, iar următoarele valori vor fi chiar valorile din șirul respectiv.
 - b) Scrieți o funcție care să afișeze o structură de date de tipul anterior.
 - c) Folosind funcția *qsort* din biblioteca *stdlib.h*, sortați crescător elementele fiecărui șir, iar apoi sortați șirurile în ordinea crescătoare a lungimilor lor.
 - d) Scrieți o funcție care să interclaseze două șiruri de numere întregi alocate dinamic și sortate crescător. Rezultatul interclasării se va reține în primul șir, iar cel de-al doilea șir va fi șters din memorie!
 - e) Folosind funcția definită anterior, interclasați toate șirurile de numere întregi date astfel încât rezultatul să fie memorat pe prima linie a structurii de date bidimensionale create la punctul a)
 - f) Adăugați la sfârșitul fișierului text *siruri.txt*, pe o linie nouă, elementele șirului obținut prin interclasarea șirurilor inițiale și apoi eliberați toată memoria alocată dinamic.

4. Fișierul text *matrice_inf.txt* conține elementele unei matrice inferior triunghiulare de numere întregi (toate elementele de deasupra diagonalei principale sunt nule) și are următoarea structură: pe prima linie un număr natural nenul n reprezentând dimensiunea matricei, pe a doua linie un număr întreg reprezentând elementul de pe prima linie a matricei, pe a treia linie două numere întregi reprezentând elementele de pe a doua linie, etc. Scrieți o funcție care citește din fișierul text *matrice_inf.txt* elementele matricei inferior triunghiulare și alocă dinamic un tablou bidimensional reprezentând matricea inferior triunghiulară. Funcția va întoarce un pointer dublu către acest tablou.
5. Aceeași cerință ca la 4 numai că avem o matrice superior triunghiulară de numere întregi (toate elementele de dedesubtul diagonalei principale sunt nule).
6. Scrieți un program care citește o matrice inferior triunghiulară și una superior triunghiulară din două fișiere text folosind funcțiile precedente și scrie rezultatul înmulțirii celor două matrice într-un fișier text.