

## ATENȚIONARE

1) Nefolosirea în rezolvare a unui proiect (format din cel puțin două fișiere – fișierele header nu se număra) și a fișierului header conduce la anularea punctajului obținut la rezolvarea celorlalte puncte din problemă. De asemenea, folosirea variabilelor globale conduce la anularea punctajului obținut.

2) Codul cu erori de compilare (inclusiv atenționări) și link-editare se notează cu 0 (zero).

3) Nefolosirea funcțiilor indicate în barem atrage o penalizare de 50% din punctaj.

4) Punctajul maxim se acordă pentru rezolvarea **CORECTĂ** a fiecărei subprobleme (punct din barem).

Livada unui fermier este organizată pe  $n$  rânduri a câte  $m$  pomi.

Fermierul are nevoie de un program cu ajutorul căruia să țină evidența producției de fructe din livadă. Fermierul vă angajează pentru a scrie programul de evidență în **C ANSI**. După discuții cu fermierul rezultă următoarele specificații necesare pentru scrierea programului:

Informațiile referitoare la livada respectivă sunt stocate într-o structură care are ca membri numărul de rânduri și numărul de pomi de pe fiecare rând și o matrice alocată dinamic prin intermediul unui pointer la pointer la real. Valoarea elementului de pe linia **i** și coloana **j** din matrice reprezintă producția de fructe (exprimată în Kg și care este un număr real) a pomului corespunzător (numărul de rânduri din livadă coincide cu numărul de linii din matrice, iar numărul de pomi de pe un rând coincide cu numărul de coloane din matrice). În această structură există de asemenea doi pointeri la caracter prin care se stochează numele livezii și tipul pomilor (meri, peri, etc.). Numele prototipului structurii va fi **LIVADA**.

De asemenea, trebuie definit tipul de date **REZULTAT** ca o uniune care conține următoarele tipuri de date: un întreg și un real.

Programul scris trebuie să realizeze următoarele operații:

1. citește de la tastatură numele livezii pentru care se va face prelucrarea (memorat prin intermediul unui pointer la caracter pentru care se alocă spațiu de memorie strict necesar);

2. citește de la tastatură tipul pomilor din livadă (memorat prin intermediul unui pointer la caracter pentru care se alocă spațiu de memorie strict necesar);

3. citește de la tastatură numărul de rânduri din livadă și numărul de pomi de pe un rând (se face validarea citirii, cele două numere trebuind să fie numere întregi pozitive; orice alte caractere introduse generând un mesaj de eroare și reluarea citirii);

4. citește de la tastatură producția de fructe realizată de fiecare pom (se face validare știind că numerele care reprezintă producția trebuie să fie numere reale pozitive; orice alte caractere introduse generând un mesaj de eroare și reluarea citirii);

5. afișează datele citite sub următoarea formă (se presupune că numele livezii este Voinești și are 6 de meri):

Livada Voinești are 6 meri, iar productia este urmatoarea:

2.3   5.4   7.5

6.5   3.1   6.0

Valorile elementelor matricei sunt afișate într-un câmp de 5 caractere cu o singură zecimală, iar valorile de pe un rând sunt separate prin 3 spații.

6. afișează un meniu care dă posibilitatea utilizatorului să aleagă o singură prelucrare din cele trei prelucrări posibile ale datelor de intrare. Prelucrările (care nu se referă la nici una din operațiile descrise mai sus) vor fi făcute prin intermediul unor funcții care au ca parametru de

intrare un pointer la o structură de tip **LIVADA** și returnează o uniune de tip **REZULTAT**. Funcțiile de prelucrare (pentru acest program) vor conține doar un mesaj care specifică prelucrarea care va fi făcută (Exemplu de mesaj: **Această funcție face prelucrarea nr. 1**). Aceste funcții au parametri de intrare **corect** indicați, iar fiecare funcție de prelucrare trebuie să returneze valoarea unei variabile de tip corespunzător.

Programul principal (funcția **main**) trebuie să conțină apelul **corect** al fiecărei funcții de prelucrare (parametri de intrare trebuie să fie corect indicați, iar valoarea returnată – dacă este cazul – trebuie să fie corect folosită).

Programul trebuie scris astfel încât să putem face prelucrări pentru mai multe seturi de date.

Alegerea prelucrării dorite se face prin folosirea instrucțiunii **switch**.

### Barem de notare

Tabel nr. 1

<b>A1.</b> Citirea informațiilor referitoare la livadă. Funcția are ca parametri doi vectori de caractere (cei doi vectori reprezintă numele livezii și tipul pomilor) și returnează o structură de tip <b>LIVADA</b> .	0,5
<b>A1a.</b> Validarea datelor citite (ca parte integrantă din <b>A1</b> )	0,2
<b>A1b.</b> Alocarea dinamică de memorie pentru un pointer la pointer (funcția primește ca parametri doi întregi și returnează un pointer la pointer la real)	0,4
<b>A1c.</b> Alocarea dinamică de memorie pentru un pointer la caracter (funcția primește ca parametru un întreg și returnează un pointer la caracter)	0,4
<b>A1d.</b> Citirea valorilor elementelor matricei și validarea valorilor citite (funcția primește ca parametri doi întregi și returnează un pointer la pointer la real)	0,4
<b>A2.</b> Afișarea informațiilor citite – după modelul dat la punctul 5 din enunț (funcția primește ca parametru un pointer la o structură LIVADA și nu returnează nimic).	0,5
<b>A3.</b> Scrierea meniului de prelucrare (funcția nu are nici un parametru și returnează un număr care reprezintă numărul opțiunii de prelucrare)	0,3
<b>A4.</b> Posibilitatea realizării prelucrărilor pentru mai multe livezi	0,3
<b>A5.</b> Folosire proiect (corect)	0,2
<b>A6.</b> Fișier header (corect și complet)	0,3
<b>A6a.</b> Definirea corectă a tipurilor de date LIVADA și REZULTAT	0,2
<b>A6b.</b> Scrierea corectă a prototipurilor celor 3 funcții de prelucrare (fiecare din cele trei funcții de prelucrare primește ca parametru un pointer la o structură de tip LIVADA și returnează o uniune de tip REZULTAT).	0,3
<b>A7.</b> Scrierea celor trei funcții de prelucrare care vor conține <b>numai câte un mesaj care indică ce prelucrare trebuie făcută</b> (cu respectarea prototipurilor indicate la punctul A6b).	0,6
<b>A8.</b> Dealocarea corectă a zonelor de memorie alocate	0,4
<b>A9.</b> Funcția main (complet – apelarea corectă a funcțiilor, folosirea meniului și utilizarea instrucțiunii switch pentru apelul <b>corect</b> al funcțiilor de prelucrare).	1,0
<b>TOTAL TABEL 1</b>	6 p