

Implementați o aplicație în limbajul C care implementează soluții la probleme de gestionare a programarilor la Serviciul Pasapoarte în vederea eliberării documentelor de calatorie.

1. Scrieți secvența de cod sursă pentru crearea unei structuri de date de tip **Lista Dublu Inlantuita** ce conține date aferente programarilor. Inserarea unei programari se implementează într-o funcție care se apelează în secvența de creare a structurii **Lista Dublu Inlantuita**. Lista dubla conține minim 10 programari incarcate in aplicatie dintr-un fisier cu date de intrare.

Structura **Programare** se va defini astfel încât să conțină minim 7 câmpuri, astfel: **cod (char*)**, **data_calendaristica (zi, luna, an)**, **ora**, **nume_solicitant (char*)**; celelalte 3 campuri sunt definite la alegere.

Cerințe de implementare:

- Definire structură **Programare**. (0,25p)
- Definire funcție de inserare nod in lista dubla cu parametri de I/O definiți complet și corect. (0,25p)
- String-urile preluate din fișier trebuie să accepte prezența simbolului **blank**. (0,25p)
- Absență memory leaks. (0,25p)
- Implementare logică de creare structură **Lista Dubla**. (0,25p)
- Gestionare structura **Lista Dubla** cu adresele primului, respectiv ultimului nod. (0,25p)
- Populare completă și corectă a structurii **Lista Dubla** cu date de intrare din fisier. (0,25p)
- Testare implementare cu afisarea la consola a continutului structurii **Lista Dubla** prin traversarea in ambele sensuri. (0,25p)

2. Scrieți și apelați funcția pentru determinarea programarilor din structura creată la cerinta 1) pe baza lunii calendaristice specificata ca parametru de intrare. Programarile identificate sunt salvate într-un vector și **NU** partajează zone de memorie heap cu structura **Lista Dubla**. Vectorul si dimensiunea acestuia se returnează în **main()** prin tipul de retur si/sau lista de parametri ai funcției.

Cerințe de implementare:

- Definire funcție cu parametri de I/O definiți complet și corect. (0,25p)
- Realizare deep-copy a programarilor în structura **Vector**. (0,25p)
- Implementare logică de determinare și salvare a programarilor în structura **Vector**. (1,00p)
- Populare completă și corectă a structurii **Vector**. (0,25p)
- Testare implementare prin apel de functie si afisare la consola a rezultatului obtinut la apel. (0,25p)

3. Scrieți și apelați funcția pentru gruparea pe luna calendaristica a programarilor din lista dubla creata la cerinta 1) prin crearea unei structuri **Tabela de Dispersie**. Cheia de cautare in tabela de dispersie este luna calendaristica. **Tabela de dispersie NU** partajeaza zone de memorie heap cu structura **Lista Dubla**. Tabela de dispersie se returnează în **main()** prin tipul de retur sau lista de parametri ai funcției.

Cerințe de implementare:

- Definire funcție cu parametri de I/O definiți complet și corect. (0,25p)
- Realizare deep-copy a programarilor în **Tabela de Dispersie**. (0,25p)
- Implementare operatie de inserare programare in **Tabela de Dispersie**. (0,25p)
- Implementare logică de creare structura **Tabela de Dispersie**. (1,50p)
- Populare completă și corectă a structurii **Tabela de Dispersie**. (0,25p)
- Testare implementare prin apel de functie si afisare la consola a rezultatului obtinut la apel. (0,25p)

4. Scrieți și apelați funcția pentru determinarea numarului maxim lunar de solicitanți pentru programarile salvate in structura **Lista Dubla** creata la cerinta 1). Se iau in considerare toate programarile efectuate pentru anul calendaristic curent. Numarul maxim, precum si luna calendaristica in care s-a inregistrat se returnează în **main()** prin tipul de retur si/sau lista de parametri ai functiei.

Cerințe de implementare:

- Definire funcție cu parametri de I/O definiți complet și corect. (0,25p)
- Calcul numar programari pe fiecare luna calendaristica a anului curent. (1,00p)
- Implementare logică de calcul a valorii maxime lunare a programarilor. (0,75p)
- Testare implementare prin apel de functie si afisare la consola a rezultatului obtinut la apel. (0,25p)

5. Scrieți și apelați funcțiile care dezalocă structurile **Lista Dubla**, **Vector** si **Tabela de Dispersie** precum și toate structurile auxiliare utilizate în implementarea cerințelor (dacă este cazul).

Cerințe de implementare:

- Definire funcții cu parametri de I/O definiți complet și corect. (0,15p)
- Absență memory leaks. (0,15p)
- Actualizare variabile de gestionare a structurilor în funcția **main()**. (0,20p)
- Implementare logică de dezalocare a structurilor de date. (0,30p)
- Testare implementare, dezalocare completă și corectă a structurilor prin apel de functii si afisare la consola a rezultatelor obtinute la apel. (0,20p)
- Absență dezalocări structuri auxiliare utilizate. (-0,20p)

MENTIUNI:

- Proiectele cu erori de compilare nu vor fi evaluate.
- Implementările nu trebuie să conțină variabile definite la nivel global sau variabile statice.
- Implementările nu trebuie să conțină structuri predefinite (ex STL, 3rd party libraries etc).
- Implementările plagiate vor fi evaluate cu 0 puncte, indiferent de sursă.
- Toate cerințele trebuie apelate și demonstrate în funcția main() pentru a fi evaluate.
- Art. 72 (1) Pentru următoarele fapte, studenții vor fi exmatriculați fără drept de reînmatriculare în Academia de Studii Economice din București:
 - (c) încercarea de promovare prin fraudă a examenelor sau a altor evaluări;