



Bilet-2 22 - Examen sdd vara 2022

Data Structures Structuri de Date (Academia de Studii Economice din București)

Implementați o aplicație în limbajul C ce rezolvă probleme specifice proiectelor de investiții.

1. Definiți structura **Proiect** ce conține: cod proiect (**unsigned int**), titlu proiect (**char***), beneficiar (**char***), număr executanți (**unsigned char**), buget alocat (**float**).

Creați o **tabelă de dispersie** cu cel puțin 5 proiecte ale căror date sunt preluate dintr-un fișier text. Cheia de căutare în tabela de dispersie este **beneficiar**. (1 p)

2. Afișați elementele din tabela creată mai sus prin traversarea întregii structuri conform mecanismului de tratare a coliziunilor. La consola de execuție se vor afișa următoarele date: **cod proiect, beneficiar**. (0,5 p)
3. Implementați funcția care returnează bugetul total de investiții al unui beneficiar specificat ca parametru de intrare al funcției. Se iau în considerare proiectele salvate în tabela de dispersie creată mai sus. Funcția implementată se apelează în funcția **main()**, iar rezultatul apelului se afișează în consola de execuție a aplicației. (1,5 p)
4. Implementați funcția care modifică denumirea unui beneficiar pentru proiectele salvate în tabela de dispersie creată mai sus. Beneficiarul a cărui denumire este modificată reprezintă parametru de intrare al funcției, împreună cu noua denumire a sa. Funcția implementată se apelează în funcția **main()**, iar tabela de dispersie se va afișa la consolă conform cerinței 2). (1,5 p)
5. Implementați funcția care salvează într-o listă simplă proiectele (din tabela de mai sus) cu un buget alocat peste un nivel specificat ca parametru de intrare al funcției. Datele sunt preluate din tabela de dispersie creată mai sus și nu sunt partajate zone de memorie heap între cele două structuri (tabelă și listă) Funcția implementată se apelează în funcția **main()**, iar rezultatul apelului (lista simplă) se afișează în consola de execuție a aplicației. (1,5 p)

Următoarele aspecte trebuie considerate cu privire la implementare:

- Implementarea **NU** determină apariția de **memory leaks**.
- Funcțiile implementate **NU** conțin apeluri de funcții standard privind operații de **I/O** cu dispozitive standard (ex: **printf, scanf** etc). Excepție face funcția **main()**.
- Funcțiile implementate care nu se apelează în funcția **main()** **NU** vor face obiectul evaluării.
- Secvențele de cod sursă comentate **NU** vor face obiectul evaluării.