

Student

pa:

Bilet #1 / (140 min)

15.06.2018

Implementați o aplicație în limbajul C ce rezolvă probleme de gestionare unei rețele de magazine de desfacere bunuri de larg consum.

1. Scrieți funcția de creare a unei structuri de tip **graf implementat prin listă de adiacență** ce conține date aferente unor magazine. Legătura între două magazine reprezintă distanța dintre acestea. Structura **Magazin** este definită astfel: id magazin (*int*), denumire magazin (*char**), localitate magazin (*char**), suprafață (*float*), număr angajați (*int*). Exemplu: {11, „La Nicusor”, „Nehoiu”, 33.69, 2} (2 p.)
2. Scrieți funcția pentru traversarea structurii listă de adiacență aferentă grafului creat la punctul 1. În consolă, se afișează toate arcele existente în graf sub forma (*i, j, w*), unde *i* este id magazin pentru vârf *i* din graf, *j* este id magazin pentru vârf *j* din graf, iar *w* este distanța dintre magazinele (*i, j*). Rezultatul execuției funcției este afișat la consolă. (1 p.)
3. Scrieți funcția care determină suprafața totală ocupată de toate magazinele din rețeaua de desfacere. Funcția se apelează și rezultatul returnat (suprafața totală) este afișat la consolă. Funcția nu include operații I/O cu tastatura/monitorul. (1 p.)
4. Scrieți funcția care determină valoarea chiriei de plată pentru fiecare magazin, returnând un vector de chirii aferent tuturor magazinelor din rețeaua de desfacere. Valoarea medie a chiriei este de **9 lei / m²**. Funcția se apelează și rezultatul returnat (vector de chirii) este afișat la consolă. Funcția nu include operații I/O cu tastatura/monitorul. (1 p.)
5. Scrieți funcția care șterge un vârf (magazin) din lista de adiacență (inclusiv muchiile din care face parte) pe baza id-ului specificat ca parametru de intrare. Rezultatul execuției funcției este validat prin apelul funcției de la punctul 2). Funcția nu include operații I/O cu tastatura/monitorul. (2 p.)
6. Scrieți funcția pentru modificarea numărului de angajați pentru un magazin specificat prin id ca parametru de intrare al funcției. Noul număr de angajați este specificat, de asemenea, prin parametru de intrare al funcției. Rezultatul execuției funcției este afișat la consolă prin apelul funcției de la punctul 2). Funcția nu include operații I/O cu tastatura/monitorul. (1 p.)
7. Scrieți funcția care traversează DF (adâncime) structura graf. Rezultatul execuției funcției (vector id-uri magazine traversate) este afișat la consolă. Funcția nu include operații I/O cu tastatura/monitorul. (2 p.)

Student: _____

Grupa _____

Bilet #2 / (140 min)

15.06.2018

Implementați o aplicație în limbajul C ce rezolvă probleme de gestionare a facturilor emise la nivelul unei firme.

1. Scrieți funcția de creare a unei structuri de tip **arbore AVL** ce conține date aferente unor facturi. Structura **Factura** este definită astfel: număr factura (**int**), data emiterii (**char***), denumire client (**char***), valoare factură (**float**), număr produse facturate (**int**). Exemplu: {4899, „2018.06.13”, „La Nicusor”, 333.62, 3} (2 p.)
2. Scrieți funcția pentru traversarea în inordine a structurii creată la punctul 1). În consolă, se afișează toate datele aferente facturii. Rezultatul execuției funcției este afișat la consolă. (1 p.)
3. Scrieți funcția pentru traversarea arborelui AVL astfel încât să fie evidențiate facturile pe fiecare nivel (număr factură). (2 p.)
4. Scrieți funcția care determină valoarea tuturor facturilor emise după o dată calendaristică specificată prin parametru de intrare al funcției. Funcția se apelează și rezultatul returnat (valoare facturi) este afișat la consolă. Funcția nu include operații I/O cu tastatura/monitorul (1 p.)
5. Scrieți funcția care determină numărul de facturi emise pentru un client specificat ca parametru de intrare al funcției. Funcția se apelează și rezultatul returnat (număr facturi) este afișat la consolă. Funcția nu include operații I/O cu tastatura/monitorul. (1 p.)
6. Scrieți funcția pentru corectarea valorii facturilor din arborele AVL prin adăugarea unui comision bancar în valoare de 2,50 lei / factură. Rezultatul execuției funcției este afișat la consolă prin apelul funcției de la punctul 2). Funcția nu include operații I/O cu tastatura/monitorul. (1 p.)
7. Scrieți funcția șterge o factură din arborele AVL. Factura este identificată prin număr, fiind specificat ca parametru de intrare al funcției. Funcția nu include operații I/O cu tastatura/monitorul. (2 p.)

Student: _____ Grupa: _____ Bilet #3 / (140 min) | 15.06.2018

Implementați o aplicație în limbajul C ce rezolvă probleme de gestionare a unor articole trimise la o conferință.

1. Scrieți funcția de creare a unei structuri de tip **Tabela de dispersie** ce conține date aferente unor articole. Structura **Articol** este definită astfel: id articol (*int*), titlu articol (*char**), categorie (*char**), număr pagini (*int*), număr autori (*int*). Exemplu: {11, „IoT Security”, „Security”, 12, 2}. Inserarea se face pe baza campului categorie al structurii Articol, iar rezolvarea cazurilor de coliziune se face prin chaining. (2 p. (1,5+0,5))
2. Scrieți funcția pentru traversarea structurii Tabela de dispersie creata la punctul 1). În consolă, se afișează toate informațiile despre articolele stocate in tabela de dispersie. Rezultatul execuției funcției este afișat la consolă. (1 p.)
3. Scrieți funcția care determină numărul total de autori din tabela de dispersie. Rezultatul execuției funcției este returnat si afișat la consolă. Funcția nu include operații I/O cu tastatura/monitorul/fișier. (1 p.)
4. Scrieți funcția care returnează un vector cu toate articolele dintr-o categorie primita ca parametru. Rezultatul execuției funcției (vector de articole) este afișat la consolă. Funcția nu include operații I/O cu tastatura/monitorul/fișier. (1 p.)
5. Scrieți funcția care șterge un articol din tabela de dispersie pe baza categoriei și a id-ului specificate ca parametri de intrare. Rezultatul execuției funcției este validat prin apelul funcției de la punctul 2). Funcția nu include operații I/O cu tastatura/monitorul/fișier. (2 p.)
6. Scrieți funcția pentru mărirea numărului de pagini cu 1 pentru toate articolele din tabela de dispersie. Rezultatul execuției funcției este afișat la consolă. Funcția nu include operații I/O cu tastatura/monitorul/fișier. (1 p.)
7. Scrieți funcția care returnează un vector cu numărul de pagini aferente fiecărei categorii. Sunt contorizate articolele din fiecare categorie. Rezultatul execuției funcției (vectorul cu numărul de pagini) este afișat la consolă. Funcția nu include operații I/O cu tastatura/monitorul/fișier. (2 p.)

Student: [REDACTED]

Grupa: [REDACTED]

Bilet #4 / (140 min)

15.06.2018

Implementați o aplicație în limbajul C ce rezolvă situații de preluare a pacienților care așteaptă intrarea în cabinetul medicului de familie.

- 1. Scrieți funcția de creare a unei structuri de tip **coadă de priorități** ce conține date aferente unor pacienți. Structura **Pacient** este definită astfel: cod pacient (**int**), nume pacient (**char***), descriere problema (**char***), vârsta (**float**), prioritate (**int**). Exemplu: {1234, "Ionescu Gigel", "Simptome de raceala", 23, 3}. Prioritățile în funcție de care se face intrarea în cabinet se definesc astfel: Categoria 1 – gravide și copii cu vârsta sub 1 an (prioritate = 5), Categoria 2 – copii cu vârsta peste 1 an (prioritate = 4), Categoria 3 – restul pacienților având priorități 1, 2 sau 3 în funcție de numărul de ore cât au așteptat la coadă. (2 p)
- 2. Scrieți funcția pentru traversarea structurii creată la punctul 1). În consolă, se afișează toate datele aferente unui pacient. Rezultatul execuției funcției este afișat la consolă. (1 p)
3. Scrieți funcția pentru inserarea unui nou pacient în coadă, funcție care primește ca parametru și categoria din care face parte noul pacient (sub forma **char* categorie**), astfel încât să îi fie atribuită prioritatea corespunzătoare. În cazul categoriei 3 se consideră implicit prioritatea cu valoarea 1. (2 p)
4. Scrieți funcția care determină numărul de gravide și copii cu vârsta sub 1 an care așteaptă la coadă pentru a intra în cabinet. Funcția se apelează și rezultatul returnat (numărul de gravide și copii cu vârsta sub 1 an) este afișat la consolă. Funcția nu include operații I/O cu tastatura/monitorul (1 p.)
5. Scrieți funcția care realizează creșterea cu o unitate a priorităților pacienților care au așteptat minim 2 ore la coadă (cu excepția gravidelor și copiilor sub 1 an și peste 1 an). Funcția se apelează și rezultatul este afișat la consolă prin apelul funcției de la punctul 2). (1 p.)
6. Scrieți funcția pentru realizarea unei statistici într-un fișier text privind problemele pentru care pacienții s-au prezentat la medic (Exemplu: Simptome de raceala – 3 pacienți, Durere de spate – 2 pacienți, Durere de cap – 1 pacient). Funcția nu include operații I/O cu tastatura/monitorul (1 p.)
7. Scrieți funcția pentru extragerea unui pacient din coada de priorități. Funcția nu include operații de I/O cu tastatura/monitorul. Funcția se apelează și rezultatul este afișat la consolă prin apelul funcției de la punctul 2). (2 p.)