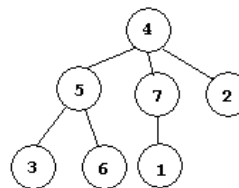


Specializarea Matematică-informatică intensiv informatică

Subiectul II (30 de puncte)

Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera care corespunde răspunsului corect.

1. Care este vectorul "de tați" pentru arborele cu rădăcină din figura alăturată? (4p.)



a. 0 0 5 7 6 5 1

b. 1 0 0 7 6 5 0

c. 7 4 5 0 4 5 4

d. 7 4 5 0 4 5 7

2. O listă liniară simplu înlănțuită, alocată dinamic, memorează în câmpul **ref** al fiecărui nod adresa următorului nod din listă sau **NULL** în cazul în care nu există un nod următor. Lista conține cel puțin 5 noduri, adresa primului nod este memorată în variabila **p**, iar variabilele **a** și **b** sunt de același tip cu **p**. Adresa cărui nod va fi memorată în variabila **b**, după executarea secvenței alăturate de program? (6p.)

```
a=p;
while (a->ref != NULL)
{ b=a;
  a=a->ref;
}
```

a. Nodul aflat în mijlocul listei

b. Penultimul nod al listei

c. Ultimul nod al listei

d. Nodul al treilea din listă

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

3. Scrieți listele de adiacență prin care este reprezentat un exemplu de graf neorientat conex, cu 6 noduri, numerotate de la 1 la 6, care este eulerian, dar **NU** este hamiltonian. (4p.)

4. Scrieți ce se afișează pe ecran în urma executării secvenței de program alăturate, în care variabila **s** memorează un șir de cel mult 12 caractere, iar variabila **i** este de tip întreg. (6p.)

```
char s[13]="abcdefghoid";
i=0;
cout<<strlen(s); | printf("%d",strlen(s));
while (i<strlen(s))
    if (strchr("aeiou",s[i])!=NULL)
        strcpy(s+i,s+i+1);
    else i++;
cout<<" "<<s; | printf(" %s",s);
```

5. Scrieți un program C/C++ care citește de la tastatură un număr natural **n** ($2 < n < 25$) și apoi construiește în memorie o matrice cu **n** linii și **n** coloane, numerotate de la 1 la **n**, ale cărei elemente primesc valori după cum urmează:

- elementele aflate pe diagonala secundară sunt toate nule;
- elementele de pe coloana **i** ($1 \leq i \leq n$), aflate deasupra diagonalei secundare, au valoarea egală cu **i**;
- elementele de pe linia **n-i+1** ($1 \leq i \leq n$), aflate sub diagonala secundară, au valoarea egală cu **i**.

Programul afișează pe ecran matricea construită, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu.

Exemplu: pentru **n=4** se va afișa matricea alăturată.

(10p.)

```
1 2 3 0
1 2 0 3
1 0 2 2
0 1 1 1
```

Subiectul III (30 de puncte)

Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Fie subprogramul `fct` definit alăturat, parțial. Inițial, variabile întregi `a`, `b` și `c` au valorile `a=8`, `b=31` și `c=9`, iar după apelul `fct(a,b,c)`, valorile celor trei variabile sunt `a=9`, `b=31` și `c=39`. Care poate fi antetul subprogramului `fct`? (4p.)
- ```
void fct(....)
{
 x=x+1;
 y=y-1;
 z=x+y;
}
```
- a. `void fct(int &x,int &y,int &z)`      b. `void fct(int x,int &y,int &z)`  
c. `void fct(int x,int y,int z)`      d. `void fct(int &x,int y,int &z)`

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. Trei băieți, **Alin**, **Bogdan** și **Ciprian**, și trei fete, **Delia**, **Elena** și **Felicia**, trebuie să formeze o echipă de 3 copii, care să participe la un concurs. Echipa trebuie să fie mixtă (adică să conțină cel puțin o fată și cel puțin un băiat). Ordinea copiilor în echipă este importantă deoarece aceasta va fi ordinea de intrare a copiilor în concurs (de exemplu echipa **Alin, Bogdan, Delia** este diferită de echipa **Bogdan, Alin, Delia**).
- Câte echipe se pot forma, astfel încât din ele să facă parte simultan **Alin** și **Bogdan**?
  - Dați exemplu de o echipă corect formată din care să nu facă parte nici **Alin** și nici **Bogdan**. (6p.)
3. Se consideră șirul `1, 2,1, 3,2,1, 4,3,2,1, ...` construit astfel: prima grupă este formată din numărul 1, a doua grupă este formată din numerele 2 și 1, iar grupa a  $k$ -a, este formată din numerele  $k, k-1, \dots, 1$ . Se cere să se citească de la tastatură un număr natural  $n$  ( $n \leq 1000$ ) și să se afișeze pe ecran cel de al  $n$ -lea termen al șirului dat.
- a) Descrieți un algoritm de rezolvare a acestei probleme, eficient din punct de vedere al timpului de executare și al spațiului de memorie, explicând în ce constă eficiența acestuia. (4p.)
- b) Scrieți programul C/C++ corespunzător algoritmului descris (6p.)
4. Se consideră subprogramul `P` care are doi parametri:
- `n`, prin intermediul căruia primește un număr natural de cel mult 9 cifre
  - `c`, prin intermediul căruia primește o cifră.
- Subprogramul va furniza tot prin intermediul parametrului `n` numărul obținut din `n` prin eliminarea tuturor aparițiilor cifrei `c`. Dacă, după eliminare, numărul nu mai conține nicio cifră sau conține doar cifre 0, rezultatul returnat va fi 0.
- a) Scrieți doar antetul subprogramului `P`. (2p.)
- b) Pe prima linie a fișierului text **BAC.IN** se găsesc, separate prin câte un spațiu, mai multe numere naturale de cel mult 9 cifre fiecare. Scrieți programul C/C++ care citește numerele din acest fișier, utilizând apeluri ale subprogramului `P` elimină toate cifrele impare din fiecare dintre aceste numere și apoi scrie în fișierul text **BAC.OUT** numerele astfel obținute, separate prin câte un spațiu. Dacă un număr din fișierul **BAC.IN** nu conține nicio cifră pară nenulă, acesta nu va mai apărea deloc în fișierul de ieșire. (8p.)
- Exemplu:** dacă fișierul **BAC.IN** conține numerele 25 7 38 1030 45127 0 35 60 15, atunci **BAC.OUT** va avea conținutul: 2 8 42 60.