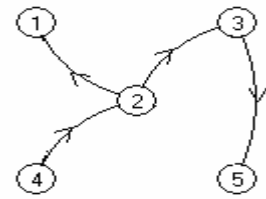


SUBIECTUL al II-lea

(30 de puncte)

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

1. Pentru ca graful orientat cu 5 vârfuri, reprezentat în figura alăturată, să devină graf tare conex, numărul minim de arce care trebuie adăugate acestuia este: **(4p.)**



- a. 1 b. 2 c. 3 d. 4
2. Se consideră arborele cu rădăcină, având 10 noduri, numerotate de la 1 la 10, reprezentat prin vectorul de „tați” $t = (2, 5, 1, 1, 0, 3, 3, 7, 4, 6)$. Rădăcina arborelui este nodul numerotat cu: **(4p.)**
- a. 0 b. 2 c. 5 d. 10

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

3. Se consideră variabila t , declarată alăturat, care memorează coordonatele, în planul xOy , ale vârfurilor A , B și C ale unui triunghi. Scrieți o instrucțiune care inițializează cu valoarea 0 coordonata x a vârfului A al triunghiului respectiv. **(6p.)**

```
type punct=record
    x, y:integer
end;
triunghi=record
    A, B, C:punct
end;
var t:triunghi;
```

4. În secvența de instrucțiuni de mai jos, variabila p este de tip întreg, iar variabila s memorează un șir de cel mult 20 de caractere, numai litere mici ale alfabetului englez. Scrieți instrucțiunile care pot înlocui punctele de suspensie astfel încât, în urma executării secvenței, să se afișeze pe ecran toate literele șirului memorat de variabila s , cu excepția vocalelor e și i . Literele se afișează în ordinea apariției lor în șir.

Exemplu: dacă șirul memorat în variabila s este *estetician*, se va afișa *sttcn*.

```
for p:=1 to length(s) do
    ....
```

(6p.)

5. Într-un tablou bidimensional, cu elemente având valori numai în mulțimea $\{0, 1\}$, numim linii „complementare” două linii cu proprietatea că oricare două elemente ale acestora, aflate pe aceeași coloană, sunt diferite. Scrieți un program **Pascal** care citește de la tastatură două numere naturale, m și n ($2 \leq m \leq 20$, $2 \leq n \leq 20$), și $m \cdot n$ valori din mulțimea $\{0, 1\}$, reprezentând elementele unui tablou bidimensional cu m linii și n coloane.

Programul afișează pe ecran numărul de linii ale tabloului care sunt „complementare” cu ultima linie a acestuia.

Exemplu: dacă $m=5$, $n=3$, pentru tabloul alăturat se afișează pe ecran valoarea 2.

(10p.)

1	0	0
1	0	1
1	0	0
0	0	1
0	1	1

SUBIECTUL al III-lea

(30 de puncte)

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

1. Algoritmul de generare a tuturor numerelor naturale formate din exact **trei** cifre, toate cifre impare, poate fi similar cu algoritmul de generare a: **(4p.)**
- a. aranjamentelor
 - b. combinațiilor
 - c. elementelor unui produs cartezian
 - d. permutărilor

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

2. Se consideră subprogramul **p**, definit alăturat. Scrieți ce valori au **p(-4)**, respectiv **p(4)**. **(6p.)**
- ```
function p(n:integer):integer;
begin
 if (n>0) and (n mod 2=0) then
 p:=1+p(n div 2)
 else p:=1
end;
```
3. Subprogramul **ordonare** are doi parametri:
- **n**, prin care primește un număr natural ( $3 \leq n \leq 20$ );
  - **a**, prin care primește un tablou unidimensional care memorează un șir de **n** numere naturale, fiecare cu cel mult **4** cifre. Cel puțin doi termeni ai șirului sunt numere pare și cel puțin unul este număr impar.
- Subprogramul modifică tabloul astfel încât termenii impari ai șirului să nu își schimbe pozițiile, iar cei pari să formeze un subșir crescător, ca în exemplu.
- Scrieți în limbajul **Pascal** definiția completă a subprogramului, precum și a tipurilor de date necesare.
- Exemplu:** dacă **n=7** și **a=(1, 4, 5, 3, 82, 6, 2)** atunci, după apel, **a=(1, 2, 5, 3, 4, 6, 82)**. **(10p.)**

4. Pentru un număr natural nenul, **n**, se consideră suma

$$s(n) = \sum_{k=1}^n \left[ \frac{n-k+1}{k!} \right] = \left[ \frac{n}{1!} \right] + \left[ \frac{n-1}{2!} \right] + \left[ \frac{n-2}{3!} \right] + \dots + \left[ \frac{1}{n!} \right]$$

în care s-a notat cu **[x]** **partea întreagă** a numărului real **x** și  $k! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot (k-1) \cdot k$

a) Scrieți un program **Pascal** care citește de la tastatură un număr natural nenul **n** ( $n < 1000$ ) și determină, utilizând un algoritm eficient din punct de vedere al timpului de executare și al memoriei utilizate, valoarea sumei **s(n)** corespunzătoare numărului citit, apoi scrie valoarea obținută în fișierul **BAC.TXT**. **(6p.)**

**Exemplu:** pentru **n=4** în fișier se scrie valoarea 5, calculată ca mai jos:

$$s(4) = \left[ \frac{4}{1!} \right] + \left[ \frac{3}{2!} \right] + \left[ \frac{2}{3!} \right] + \left[ \frac{1}{4!} \right] = 4 + 1 + 0 + 0 = 5$$

b) Descrieți succint, în limbaj natural (3-4 rânduri), algoritmul utilizat la punctul a) și justificați eficiența acestuia. **(4p.)**