VARIANTA 3

SUBIECTUL 1 (20p)

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

1. Indicaţi expresia care are valoarea true/1 dacă şi numai dacă numărul întreg memorat în variabila x are exac trei cifre. (4p.)

	Limbajul Pascal	Limbajul C/C++
a.	x<1000	x<1000
b.	(100<=x) and (x<=999)	(100<=x) && (x<=999)
c.	(-999<=x) and (x<=999)	(-999<=x) && (x<=999)
d.	(-999<=x) and (x<=-100) or (100<=x) and (x<=999)	(-999<=x) && (x<=-100) (100<=x) && (x<=999)

2. Se consideră subprogramul f cu definiția următoare: (4p.)

Limbajul Pascal	Limbajul C/C++
<pre>procedure f(x:longint); begin</pre>	<pre>void f(int x) {</pre>

Indicati valoarea f (12345)

, , , ,			
a. 5432#***	b. ****#5432	c. 5432####*	d. #***5432

3. În declararea următoare, câmpurile **x** și **y** ale înregistrării pot memora coordonatele carteziene ale unui punct din planul **xOy**. Indicați expresia în limbajul **Pascal | C/C++** care să aibă valoarea **true**|1 dacă și numai dacă punctul memorat în variabila **A** este situat pe axa **Ox**, iar punctul memorat în variabila **B** este situat pe axa **Oy**. **(4p.)**

Limbajul Pascal	Limbajul C/C++	
<pre>type punct=record x,y:real end; var A,B:punct;</pre>	<pre>struct punct { float x,y; } A,B;</pre>	
a. (A.y=0) and (B.x=0) b. (A.y=0) or (B.x=0) c. (A.x=0) or (B.y=0) d. (A.x=0) and (B.y=0	a. (A.y==0) and (B.x==0) b. (A.y==0) or (B.x==0) c. (A.x==0) or (B.y==0) d. (A.x==0) and (B.y==0	

4. Care este numărul maxim de muchii pe care îl poate avea un graf neorientat cu 5 noduri? (4p.)				
a. 4	b . 5	c . 8	d. 10	
5. Se consideră un arbore cu rădăcină reprezentat prin următorul vector "de taţi": (0,1,1,1,4,4,6,7). Câte lanţuri elementare distincte, de lungime maximă, care au extremitatea finală în nodul 8, există în arborele specificat. (4p.)				
a . 2	b. 1	c . 3	d. 0	

SUBIECTUL 2 (40p)

1. Se consideră algoritmul următor reprezentat în pseudocod:

S-a notat cu x%y restul împărțirii numărului întreg x la numărul întreg nenul y și cu [a] partea întreagă a numărului real a.

```
citeşte x,y (numere naturale)
p←1
cât timp y>0 execută
| t←y
| cât timp t>9 execută
| | t←[t/10]
| L
| cdacă x*10 = t atunci p←p+1
| altfel scrie p
| p←1
| L
| x←y
| citește y (număr natural)
L
| scrie p
```

- a) Scrieți ce se va afișa dacă se citesc, în această ordine, numerele:
 - 12 23 34 592 13 39 91 1 0? (6p.)
- b) Scrieţi un set de date de intrare pentru care algoritmul, în urma executării, va afișa valoarea 1234. (6p.)
- Scrieți în pseudocod un algoritm, echivalent cu cel dat, în care să se înlocuiască structura cât timp t>9 execută... cu o structură repetitivă de alt tip. (6p.)
- d) | Scrieti programul Pascal/C/C++ corespunzător algoritmului dat. (10p.)
- 2. Utilizând metoda backtracking se generează în ordine crescătoare toate numerele naturale de câte cinci cifre din mulţimea A={1,2,3,4}, numere în care suma primelor două cifre este 5 .Primele numere generate sunt, în ordine: 14111, 14112, 14113, 14114, 14121, 14122,... Scrieţi câte dintre numerele generate încep şi se termină cu cifra 3? (6p.)
- 3. Variabila S memorează un șir cu maximum 20 de caractere. Scrieți ce se afișează în urma executării secvenței: **(6p.)**

Limbajul Pascal	Limbajul C/C++
S:= '3a+b=9';	char S[20]="3a+b=9";
S:=copy(S,2,5);	strcpy(S,S+1);
delete (S,2,2);	strcpy(S+1,S+3);
write (S);	cout< <s;< td=""></s;<>

SUBIECTUL 3 (30p)

- 1. Scrieți definiția completă a unui subprogram sub, cu trei parametri, care primește prin intermediul parametrilor:
 - x, un număr natural, având cel mult patru cifre;
 - n, un număr natural nenul (0 < n < 100);
 - v, un tablou unidimensional care memorează un şir de n numere naturale, fiecare având cel mult patru cifre;

Subprogramul \mathbf{sub} determină dublarea fiecărei apariții a valorii parametrului \mathbf{x} în șirul de numere primit prin parametrul \mathbf{v} . De asemenea subprogramul va furniza prin intermediul parametrului \mathbf{n} numărul de numere din șirul modificat, iar prin parametrul \mathbf{v} tabloul modificat. Dacă valoarea parametrului \mathbf{x} nu apare în șirul de numere atunci valorile parametrilor nu se vor modifica.

Exemplu. Pentru valorile n=10, v=(51, 19, 3, 52, 19, 215, 19, 19, 4, 65), x=19 ale parametrilor, în urma apelului, subprogramului sub va furniza prin parametrii indicaţi valorile n=14 şi v=(51, 19, 19, 3, 52, 19, 19, 215, 19, 19, 19, 19, 4, 65).

(10p.)

- 2. Să se construiască o matrice cu n linii și m coloane care conține:
 - pe prima coloană, de sus în jos, toate numerele naturale de la 1 la n, în ordine crescătoare;
 - pe prima linie, de la stânga la dreapta toate numerele naturale de la 1 la ${\tt m}$, în ordine crescătoare;

Restul elementelor se vor inițializa cu suma elementelor vecine situate în stânga și deasupra elementului curent, ca în exemplu.

Scrieţi programul care citeşte de la tastatură două valori naturale \mathbf{n} și m (2<n<25, 2<m<25), construieşte matricea conform cerinței și o afișează pe ecran, pe linii, cu spaţii între elementele de pe fiecare linie.

De exemplu, dacă se citește de la tastatură n=3 și m=5, se va construi și se va afișa pe ecran matricea:

```
1 2 3 4 5
2 4 7 11 16
3 7 14 25 41
(10p.)
```

- 3. Fişierul text **BAC.TXT** conţine un şir **s** de cel mult un milion de numere naturale, formate fiecare din cel mult **9** cifre, separate prin câte un spaţiu
 - a) Scrieţi un program C/C++ care citeşte numerele din fişier şi determină, utilizând un algoritm eficient din punct de vedere al memoriei utilizate şi al timpului de executare, lungimea secvenţei obţinute prin eliminarea din cele două extremităţi ale şirului s a unui număr minim de numere, fără

a schimba ordinea celorlalte numere, astfel încât secvența rezultată să înceapă cu un număr format doar din cifre pare si să se termine cu un număr format doar din cifre impare. Programul va afișa pe ecran lungimea secvenței obținute.

De exemplu, dacă fișierul BAC.TXT conține numerele:

132 214 62 34 28 34 8 45 18 72 35 12 17 34 4 135 63 81 101

pe ecran se va afișa numărul 14, deoarece secvența căutată se obține prin eliminarea numerelor subliniate și este formată din 14 numere (8p.)

b) Descrieți în limbaj natural algoritmul proiectat justificând eficiența acestuia. (2p.)

Observaţii:

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul efectiv de lucru este de 3 ore.
- În rezolvările cerute, identificatorii utilizați trebuie să respecte precizările din enunț (bold), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară.