

Filieră teoretică, profil real, specializare științe ale naturii

SUBIECTUL I**(20 de puncte)**

Pentru fiecare dintre itemii de la 1 la 5, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 4 puncte.

1. Care dintre expresiile C/C++ de mai jos este echivalentă cu expresia următoare

!((a<2025)&&(b>2024))

a. **(a>=2024)&&(b<=2025)**

b. **!(a<2024) || !(b>2025)**

c. **!(a<2024) && !(b>2025)**

d. **!(a<2024) && !(b>2025)**

2. În secvența de mai jos, toate variabilele sunt de tip întreg

for(i=1;i<=5;i++)

{ for(j=1;j<=5;j++)

cout<<;

cout<<endl;

}

a. i+j%3

b. (i+j)%5

c. (i+j)%3

d. i+j%2

2 0 1 2 0

0 1 2 0 1

1 2 0 1 2

2 0 1 2 0

0 1 2 0 1

3. Indicați valoarea expresiei C/C++ alăturate.

floor(2.2)+ceil(3.7)

a. 3

b. 0

c. 1

d. 2

4. Tabloul unidimensional A are elementele: A=(2,20,27,36,50), iar în urma interclasării lui crescătoare cu tabloul unidimensional B se obține tabloul cu elementele (2,3,5,8,20,27,36,45,50,63). Indicați elementele tabloului B, în ordinea apariției lor în acesta.

a. (1,-15,-19, 9, 13)

b. (2,20,27,36,50,3,5,8,45,63)

c. (63,45,8,5,3)

d. (3,27,45,63)

5. Care dintre următoarele secvențe atribuie variabilei întregi **a** valoare **2024ⁿ**, știind că variabilele **i** și **n** sunt de tip întreg cu **n>0**?

//s1

//s2

//s3

//s4

a=1;

a=1;

a=1;

a=1;

for(i=1;i<=2024;i++)

for(i=1;i<=n;i++)

for(i=1;i<=n;i++)

for(i=1;i<=n;i++)

a=a*n;

a=a*2024;

a=n*2024;

a=a*I;

a. s1

b. s2

c. s3

d. s4

SUBIECTUL al II-lea**(40 puncte)**

1. Se consideră algoritmul alăturat descris în pseudocod:

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

a. Scrieți numărul afișat dacă pentru variabila **n** se citește valoarea **74589**. **(6p.)**

b. Scrieți toate numerele de exact trei cifre care pot fi citite pentru variabila **n** astfel încât, pentru fiecare dintre acestea, numărul afișat în urma executării algoritmului să fie **24**. **(6p.)**

c. Scrieți programul C/C++ corespunzător algoritmului dat.

(10p.)

a. Scrieți în pseudocod un algoritm, echivalent cu cel dat, în care să se înlocuiască structura **repetă...până când** cu o structură repetitivă de alt tip. **(6p.)**

citește **n** (număr natural nenul)

m ← 0

repetă

c ← n%10 ; n ← [n/10]

dacă c>5 atunci

c ← [c/2]

■

m ← m*10+c

până când n=0

scrie m

2. Variabilele **l**, **z** (numere naturale nenule) reprezintă data (luna, zi) la care expiră un produs alimentar aflat într-un depozit. Cunoscând ziua și luna curentă (**lc**, **zc**), scrieți o secvență de instrucțiuni C/C++ care să afișeze mesajul “aliment expirat”, dacă acesta este expirat sau mesajul “aliment neexpirat”, în caz contrar. **(6p.)**

3. Se consideră un tablou unidimensional în care elementele sunt, în această ordine, (37,29,17,15,11,5,2). Pentru a afla indicele elementului din tablou cu valoarea $x=17$, se aplică metoda căutării binare. Scrieți succesiunea corectă de elemente a căror valoare se compară cu valoarea lui x pe parcursul metodei indicate. (6p.)

SUBIECTUL al III-lea

(30 puncte)

1. Se consideră a și b , două valori naturale nenule $a \leq b < 10^9$. Algoritmul citește a și b și afișează, separate prin câte un spațiu, în ordine descrescătoare, toate numerele cu număr impar de divizori din intervalul $[a, b]$ sau -1 dacă nu există astfel de numere.

Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate și precizați care sunt datele de intrare și care sunt datele de ieșire.

Exemplu: dacă $a=10$ și $b=100$ algoritmul va afișa 100 81 64 49 36 25 16

(10p.)

2. Se consideră un tablou unidimensional cu maxim 255 numere naturale. Scrieți un program C/C++ care citește un astfel de tablou, modifică fiecare element astfel: prima cifră se micșorează cu o unitate iar ultima cifră se mărește cu o unitate, dacă se poate, fără a afecta celelalte cifre iar în caz contrar rămâne așa cum este. În final se afișează tabloul obținut

(10p.)

Exemplu: pentru tabloul: 25 349 150 641 1255 269
se va afișa 16 249 151 542 1256 169

3. Fișierul **numere.in** memorează cel puțin 3 numere și cel mult 10000 de numere naturale cu cel mult patru cifre reprezentând înălțimile vârfurilor munților Carpații din România. Se cere să se afișeze pe ecran, în ordine crescătoare, înălțimile celor mai înalte două vârfuri separate prin câte un spațiu.

Pentru determinarea vârfurilor cerute se utilizează un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

Exemplu: dacă fișierul **numere.in** conține numerele 2520 1659 2044 2136 1959 1878 2563 1849 2503, se afișează pe ecran 2520 2563.

a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia.

(2p.)

b) Scrieți programul C/C++ corespunzător algoritmului descris.

(8p.)