

MODEL TEST 6

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.
- Identificatorii utilizați în rezolvări 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). Datele de intrare se consideră corecte, validarea lor nefiind necesară.

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. Variabila x memorează un număr natural de exact 6 cifre. Care dintre următoarele instrucțiuni C/C++ elimină din x cele două cifre situate la mijlocul numărului?

- a. $x/10000*100+x\%100$; b. $x=x/100-x/100\%100+x\%100$;
c. $x=x/10000+x\%100$; d. $x\%10000+x/100$;

2. Care este rezultatul evaluării expresiei C/C++ următoare: $14/3*4*3/4$?

- a **12.99**
b **13.99**
c **12**
d **14**

3. Ce va afișa pe ecran următorul program?

```
#include <iostream>
using namespace std;
int main()
{int n=10, i;
 int a[]={1,22,33,4,5,6,7,8,9,11};
 for (i=n/2; i; i--) a[i+n/2-1]=a[i];
 for (i=0; i<n; i++) cout<<a[i];
 return 0;
}
```

- a. 1223345678911 b. 12233452233456 c. 12233451223345 d. 22334561223345

4. Fie secvența:

```
for (i = 0; i < 20; i++) a[i] = i * i - i;
```

Câte din primele 20 de componente ale vectorului a memorează valori care au cifra unităților 0?

- a **6**
b **7**
c **8**
d **9**

5. Se consideră secvența de mai jos, în care toate variabilele sunt de tip întreg.

```
i=2;
while(...){if(x%i==0) cout<<i<<" ";
i=i+1;}
```

Pentru a afișa în ordine crescătoare toți divizorii pozitivi ai numărului natural nenul memorat în variabila x , cu excepția lui 1 și a numărului respectiv, o expresie care poate înlocui punctele de suspensie este:

- a $i > 0$
- b $i >= 2$
- c $i <= x/2$
- d $i <= x$

SUBIECTUL al II-lea
(40 de puncte)
1. Algoritmul alăturat este reprezentat în pseudocod.

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

a. Scrieți valoarea afișată în urma executării algoritmului dacă se citește numerele **1983** și **237**.

Sc

(6)

p.)

b. Scrieți o pereche de numere care poate fi citită, astfel încât în urma executării algoritmului, să se afișeze valoarea

Sc

11.

(6)

p.)

c. Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**

d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, care să înlocuiască a doua structură repetitivă cu o structură repetitivă alt de alt tip.

(6p.)

citește a,b (numere naturale)

$x \leftarrow 0$

execută

$x \leftarrow x * 10 + a \% 10 + b \% 10$

$a \leftarrow [a/10]$

$b \leftarrow [b/10]$

cât timp $!(a=0 \text{ sau } b=0)$

$n \leftarrow 0$

repetă

$n \leftarrow n * 10 + x / 10 \% 10$

$x \leftarrow [x/100]$

până când

$x=0$

scrie n

1. În Se consideră două puncte în plan **A** și **B** de coordonate **xa, ya**, respectiv **xb, yb**. Să se scrie o condiție care are valoarea **1** dacă și numai dacă segmentul **AB** este paralel cu axa **OX**. (6p)

2. În secvența de instrucțiuni de mai jos, variabilele **i** și **j** sunt de tip întreg. Scrieți secvența alăturată, înlocuind punctele de suspensie astfel încât, în urma execuției secvenței obținute, să se afișeze pe ecran, în această ordine, numerele de mai jos. **(6p.) 2 4 6 8**

2 4 6

2 4

2

for(i=1;i<=4;i++)

{for(.....)

cout<<2*j<<' ';

cout<<endl;

SUBIECTUL al III-lea**(30 de puncte)**

1. Se citesc de la tastatură două numere naturale **n** și **k**. Să se obțină numerele **a** și **b**, unde **a** reprezintă numărul maxim extras din radicalul de ordin **k** al numărului **n**, iar **b** este numărul minim rămas sub radicalul de ordin **k**.

Scrieți în pseudocod algoritmul de rezolvare pentru problema enunțată.

Exemplu: dacă **n=2020** și **k=2**, atunci $\sqrt[2]{2020} = 2\sqrt{505}$, adică **a** va fi **2**, iar **b** va fi **505**.

(10 p.)

2. Scrieți un program Pascal/C/C++ care citește de la tastatură numere naturale din intervalul $[1, 10^9]$, în această ordine **n**, apoi cele **n** elemente ale unui tablou unidimensional. Numărul **n** aparține intervalului $[1, 10^3]$. Programul modifică ordinea valorilor din tablou, astfel încât valorile impare să fie ordonate descrescător, valorile pare să fie ordonate crescător, și fiecare număr de tip par/impar își va păstra indicii originali.

Exemplu: **n=8**, și vectorul: **2, 0, 1, 9, 2, 0, 2, 0**

Vom afișa: **0 0 9 1 0 2 2 2**,

Se observă ca numerele impare si-au păstrat indicii originali, la fel si cele pare.

(10 p.)

3. Fișierul **cifre.in** conține pe prima linie cel mult **10⁶** cifre. Să se citească cifrele din fișier și să se rearanjeze dacă este posibil astfel încât să formeze cel mai mare număr palindrom, sau mesajul **IMPOSIBIL**. Afișarea se va face pe ecran. Proiectați un algoritm eficient din punct de vedere al spațiului de memorie și al timpului de executare.

Exemple:

Cifre.in	2 0 1 9 2 0 2 0	2 2 1 2 0 2 0
Pe ecran	IMPOSIBIL	2201022

- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2 p.)**
- b. Scrieți programul Pascal/C/C++ corespunzător algoritmului proiectat. **(8 p.)**

REZOLVARE SUBIECTE
TEST 6
SUBIECTUL I
(20 de puncte)

1b	2c	3b	4c	5c	5x4p
----	----	----	----	----	------

SUBIECTUL al II-lea
(40 de puncte)

1. a. După execuția primei structuri repetitive vom obține $x=1121$

După execuția celei de-a doua structuri repetitive vom obține $n=21$

Răspuns 21

a. Orice pereche de numere care să furnizeze după execuția primei structuri repetitive valoarea x de forma $a1b1$ va da răspunsul corect ($0 \leq a, b \leq 9$). O astfel de pereche este: $a=2131$ și $b=5050$

b. Programul C/C++:

```
#include <iostream>
using namespace std;
int main()
{
    int a,b, x, n=0;
    cin>>a>>b;
    x=0;
    do{
        x=x*10+a%10+b%10;
        a=a/10;
        b=b/10;
    }while(a!=0&& b!=0);
    if(x!=0)
    do{
        n=n*10+x/10%10;
        x=x/100;
    }while( x!=0);
    cout<<n;

    return 0;}
```

d. Algoritmul în pseudocod echivalent care înlocuiește a doua structură repetitivă este:
citeste a,b (numere naturale)

$x \leftarrow 0$

```
execută
    x ← x*10 + a%10 + b%10
    a ← [a/10]
    b ← [b/10]
```

■ cat timp $a \neq 0$ și $b \neq 0$

$n \leftarrow 0$

```
cat timp  $x \neq 0$  executa
    n ← n*10 + x/10%10
    x ← [x/100]
```

scrie n

2. Condiția care rezolvă cerința: $ya == yb$

3. for (i=1; i<=4; i++)

```
{
    for (j=1; j<=5-i; j++)
        cout<<2*j<<' ';
    cout<<endl;}
```

SUBIECTUL al III - lea

(30 de puncte)

1.

```

Citește n,k
///descompunem in factori primi
a←1
b←1
d←2
cat timp(n>1) executa
|   e←0;
|   cat timp(n%d=0) executa
|   |   e←e+1
|   |   n←[n/d]
|   |   ┌─┐
|   |   │  │
|   |   └─┘
|   |   /// modificarea lui a
|   |   Pentru i←1,[e/k] executa
|   |   |   a←a*d
|   |   |   ┌─┐
|   |   |   │  │
|   |   |   └─┘
|   |   |   /// modificarea lui b
|   |   |   Pentru i←1,e%k executa
|   |   |   |   b←b*d
|   |   |   |   ┌─┐
|   |   |   |   │  │
|   |   |   |   └─┘
|   |   |   |   d←d+1
|   |   └─┘
|   ┌─┐
|   │  │
|   └─┘
Scrie a, ' ',b

```

2.

```

#include<fstream>
#include<iostream>
using namespace std;
int n,a[1002],i,j;
int main()
{
    cin>>n;
    for(i=1;i<=n;i++)
        cin>>a[i];
    ///metoda selectiei directe cu testare de paritati
    for(i=1;i<=n;i++)
        for(j=i+1;j<=n;j++)
            if (a[i]%2!=0 and a[j]%2!=0)///ambele impare-sa fie descrescatoare
            {
                if (a[i]<a[j])
                    swap(a[i],a[j]);
            }
            else
            if (a[i]%2==0 and a[j]%2==0)///ambele pare-sa fie crescatoare
            {
                if (a[i]>a[j])
                    swap(a[i],a[j]);
            }
        }
    for(i=1;i<=n;i++)
        cout<<a[i]<<' ';
}

```

3.

a. Fiind doar cifre în fișierul de intrare, vom defini un vector de frecvență de dimensiune 10. Problema nu va avea soluție dacă există 2 sau mai multe cifre cu frecvențe impare. Dacă există soluție, cifrele se vor afișa în două seturi, folosind în fiecare set un număr de cifre egal cu jumătate din frecvența fiecărei cifre. Palindromul de valoare maximă va începe cu un set de cifre alese în sens descrescător, va continua cu cifra cu frecvență impară (dacă există) și se va termina cu al doilea set de cifre alese în sens crescător.

```

#include<fstream>
#include<iostream>
using namespace std;
ifstream fin("cifre.in");
int x,f[10],c,imp,i;
int main()
{
    while(fin>>x)
        f[x]++;
    imp=0;
    for(c=0;c<=9;c++)
        if (f[c]%2==1) imp++;
    if (imp>1) cout<<"IMPOSIBIL";
    else
    {
        //afisam cifrele descrescator jumatate din frecventa..
        for(c=9;c>=0;c--)
            for(i=1;i<=f[c]/2;i++)
                cout<<c;
        //verificam daca avem o frecv impara
        for(c=0;c<=9;c++)
            if (f[c]%2==1) cout<<c;
        //afisam cifrele crescator jumatate din frecventa..
        for(c=0;c<=9;c++)
            for(i=1;i<=f[c]/2;i++)
                cout<<c;
    }
}

```

BAREM DE EVALUARE ȘI DE NOTARE**TEST 6**

- Se punctează oricare alte modalități de rezolvare corectă a cerințelor.
- Nu se acordă punctaje intermediare, altele decât cele precizate explicit prin barem. Nu se acordă fracțiuni de punct. Se acordă 10 puncte din oficiu. Nota finală se calculează prin împărțirea punctajului total acordat pentru lucrare la 10.
- Utilizarea unui tip de date care depășește domeniul de valori precizat în enunț este acceptată dacă acest lucru nu afectează corectitudinea în funcționarea programului.

SUBIECTUL I**(20 de puncte)**

Răspuns	Punctaj
1b 2c 3b 4c 5c	5x4p.

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

1.	a) Răspuns corect: 21	6p.	
	b) Răspuns corect: orice pereche de numere care furnizeaza rezultatul 11.	6p.	
	c) Pentru program corect - declarare variabile - citire date - afișare date - instrucțiuni repetitive (*) - atribuiri - corectitudine globală a programului ¹⁾	10p. 1p. 1p. 1p. 5p. 1p. 1p.	(*) Se acordă numai 3p. dacă doar una dintre instrucțiunile repetitive este conform cerinței.
	d) Pentru algoritm pseudocod corect - echivalență a prelucrării realizate, conform cerinței (*) - corectitudine globală a algoritmului ¹⁾	6p. 5p. 1p.	(*) Se acordă numai 2p. dacă algoritmul are a doua structură repetitivă conform cerinței, principial corectă, dar nu este echivalent cu cel dat.
2.	Pentru rezolvare corectă ay==by	6p.	
3.	Pentru rezolvare corectă - acces la elementele fiecărei linii - atribuire a valorilor indicate elementelor conform valorilor afișate (*) - corectitudine globală a secvenței ¹⁾	6p. 1p. 4p. 1p.	(*)Se acordă câte 1p. pentru atribuirea valorilor conform cerinței, pentru elementele situate pe fiecare linie.

¹⁾ Corectitudinea globală vizează structura, sintaxa, alte aspecte neprecizate în barem.

SUBIECTUL al III - lea**(30 de puncte)**

1	Pentru algoritm corect - citirea datelor - determinarea numerelor cerute (*) - scrierea datelor - scriere principial corectă a structurilor de control (**)	10p. 1p. 6p. 1p. 2p	(*) Se acordă câte 2p. pentru fiecare aspect al cerinței (factori primi și exponenți + modificare variabila a + modificare variabila b) (**) Se acordă punctajul pentru orice formă corectă de structură repetitivă sau decizională.
2	Pentru program corect - declarare a unei variabile care să memoreze un tablou - citire a datelor - memorare a numerelor conform cerinței(*) - afisarea tabloului - declarare a variabilelor simple, corectitudine globală a programului)	10p. 1p. 1p. 6p. 1p. 1p	(*) Se acordă cele 6pct. astfel: - 3p prezentarea metodei de sortare (sel. Directa) - 1p verificare valori impare - 1p verificare valori pare - 1p realizarea interschimbării
3	a) Pentru răspuns corect - coerență a descrierii algoritmului (*) - justificare a elementelor de eficiență	2p. 1p. 1p.	(*) Se acordă punctajul chiar dacă algoritmul ales nu este eficient. (**) Se acordă numai 3p. dacă algoritmul este principial corect, dar nu oferă rezultatul cerut pentru
	b) Pentru program corect - operații cu fișiere: declarare, pregătire în vederea citirii, citire din fișier - determinare a valorii cerute (*),(**) - utilizare a unui algoritm eficient (***) - declarare a variabilelor, afișare a datelor, corectitudine globală a programului)	8p. 1p. 5p. 1p. 1p.	toate seturile de date de intrare. (***) Se acordă punctajul numai pentru un algoritm liniar care utilizează eficient memoria. O soluție posibilă este folosirea vectorilor de frecvență. Cazul IMPOSIBIL , e îndeplinit dacă avem 2 sau mai multe cifre cu frecvențe impare. Soluția va fi obținută prin afișarea descrescătoare a cifrelor și afișarea crescătoare a cifrelor, în fiecare parte punând jumătate din câte am citit. Caz special ce cifra va fi la mijloc, dacă avem 1 singură frecvență impară?