

MODEL TEST 4

- 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. Precizați care din expresiile de mai jos este adevărată pentru **a** cel mai mare număr natural de patru cifre distincte și **b** cel mai mic număr natural de patru cifre distincte.

- a. $(a/b==8)\|(a\%b==0)$
c. $(a\%b==8)\|(a/b==0)$
- b. $(a/b==9)\&\&(a\%b>0)$
d. $(a\%b==9)\&\&(a/b>0)$

2. Fie tabloul unidimensional (**20, 15, 12, 8, 4, 2, 1**). Pentru a verifica dacă numărul 5 se află printre elementele tabloului, se aplică metoda căutării binare. Care este succesiunea corectă de elemente cu care se compară numărul căutat?

- a. 1,2,4,8 b. 8,4,2 c. 1,2,4,8,12,15,20 d. 8,2,4

3. Variabilele **x** și **y** sunt de tip real. O transcriere în limbajul C/C++ a expresiei alăturate este: $\sqrt{(x^2/y^x)}$

- a. `sqrt(pow(x,2)/y*x)`
b. `pow(x*x,sqrt(x,y))`
- c. `sqrt((x*x)/pow(y,x))`
d. `sqrt(pow(x,x)/pow(y,x))`

4. Pentru un număr natural **n** de cel mult 9 cifre, care din următoarele secvențe de cod calculează inversul acestuia?

- I)
`m=0;`
`for(i=0;i<=n;i++)`
`m=m+n/10;`
- II)
`i=1;`
`m=0;`
`while(i<=n){`
`m=m*10+(n/i)%10;`
`i=i*10;}`
- III)
`m=0;`
`while(n>0)`
`{m=m*10+n%10;`
`n=n/10;`
`}`

- a. I și II b. II c. III d. II și III

5. Elementele unui tablou unidimensional sunt, în ordine, (4,8,1,3,9,10,12). Asupra acestuia se aplică algoritmul de sortare prin metoda bulelor. Care este conținutul tabloului la sfârșitul celei de a treia iterație a algoritmului?

- a. 4 1 3 8 9 10 12 b. 1 4 3 8 9 10 12 c. 1 3 4 8 9 10 12 d. 4 1 8 3 9 10 12

Subiectul II**(40 de puncte)**

1. În algoritmul alăturat reprezentat în pseudocod, s-a notat cu $x\%y$ restul împărțirii numărului natural x la numărul natural nenul y și cu $[x]$ partea întregă a numărului real x .

- Scrieți valoarea afișată dacă se citesc valorile 51467 și 67. **(6p)**
- Dacă pentru b se citește valoarea 73, scrieți numărul valorilor de 4 cifre ce pot fi citite pentru a astfel încât valoarea afișată să fie divizibilă cu 50. **(6p)**
- Scrieți un algoritm echivalent cu cel dat care să utilizeze o structură repetitivă cu test final. **(6p)**
- Scrieți programul C/C++ corespunzător algoritmului dat **(10p)**

citește a, b (numere naturale) $p \leftarrow 1$ cât timp $a * b > 0$ și $a \% 10 = b \% 10$ execută $a \leftarrow [a/10]$ $b \leftarrow [b/10]$ $p \leftarrow p * 10$ $a \leftarrow a * p$ scrie a

2. O firmă de agenți de vânzări oferă un bonus de 5% (din vânzările efectuate de agent) dacă vânzările depășesc valoarea de X lei pe lună, sau 10% dacă vânzările acestuia depășesc Y lei pe lună. Scrieți instrucțiunile prin care se calculează în variabila reală b valoarea bonusului, știind că vânzările totale ale unui agent sunt memorate în variabila reală v .

(6p)

3. Radu deține un restaurant, iar ca să mărească vânzările, a decis să efectueze și livrări la domiciliu; la sfârșitul zilei, vrea să afle care a fost valoarea medie a unei comenzi livrate la domiciliu. Știind că numărul total de comenzi livrate la domiciliu este memorat în variabila de tip întreg n , valoarea totală a vânzărilor din acea zi este memorată în variabila t , iar valoarea totală a comenzilor livrate la domiciliu este memorată în variabila v , declarați cele 3 variabile și scrieți instrucțiunea care calculează și afișează valoarea medie a comenzilor livrate la domiciliu, rotunjită la întregul de sus. Variabilele t și v sunt de tip real.

(6p)**Subiectul al III – lea****(30 de puncte)**

- Scrieți un program C/C++ care citește de la tastatură a și b două numere naturale cu cel mult 9 cifre fiecare și afișează pe ecran cel mai mare număr cu cifrele impare distincte care apar atât în a cât și în b sau 0 dacă nu există astfel de cifre.

Exemplu: dacă $a=45013$ și $b=215578$ atunci se va afișa valoarea 51.

(10 p)

- Scrieți un program C/C++ care citește de la tastatură numere naturale din intervalul $[0, 10^9]$, în această ordine: numerele m , p și n , apoi cele n elemente ale unui tablou unidimensional de maxim 100 elemente. Programul modifică ordinea valorilor din tablou, astfel încât ultimele m poziții să fie ocupate de mulțimea formată din elemente ale tabloului divizibile cu p , iar primele $n-m$ poziții să fie ocupate de mulțimea celorlalte numere, într-o ordine oarecare.

Valorile tabloului vor fi afișate pe ecran, separate prin câte un spațiu. Dacă tabloul nu conține cel puțin m valori divizibile cu p , următoarea linie va afișa numărul de valori divizibile cu p , găsite.

Exemplul 1: $m=3$, $p=2$, $n=12$ și tabloul (21, 5, 18, 3, 7, 4, 10, 55, 40, 22, 19, 33) unul din șirurile afișate poate fi: (21, 5, 33, 3, 7, 19, 22, 55, 40, 10, 4, 18)

Exemplul 2: $m=5$, $p=5$, $n=12$ și tabloul (21, 5, 18, 3, 7, 4, 10, 55, 40, 22, 19, 33) unul din șirurile afișate poate fi (21, 33, 18, 3, 7, 4, 19, 22, 40, 55, 10, 5) apoi, pe rândul următor, valoarea 4.

(10 p)

3. Fișierul **BAC.IN** conține cel mult un milion de numere naturale de cel mult 9 cifre fiecare separate prin câte un spațiu. Să se afișeze pe ecran cel mai mare număr **a** de exact 3 cifre care nu apare în fișier pentru care există un număr **b**, care apare în fișier, astfel încât $a+b=1000$.

Dacă nu există niciun astfel de număr se va afișa mesajul **NU EXISTA**.

Proiectați un algoritm eficient din punct de vedere al timpului de executare și justificați eficiența acestuia.

Exemple:

BAC.IN	22 8425 4567 6 8 999 33 995 800 200 45123	8425 994 4567 5 6 994 3300 995 800 200 45123
Pe ecran	994	NU EXISTA

(10 p)

REZOLVARE SUBIECTE
TEST 4

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.

Subiectul I

1. b	2. d	3. c	4. d	5. c
------	------	------	------	------

Subiectul II

1. a. 51400

b. 450 de valori (toate numerele de forma $\overline{xy73}$, $\overline{xy03}$, $\overline{xy53}$, $\overline{xy50}$, $\overline{xy00}$, $x \in \{1,2,\dots,9\}$, $y \in \{0,1,\dots,9\}$)

c. citește a,b (numere naturale)
 p←1
 dacă a*b>0 și a%10=b%10 atunci
 -repetă
 a←[a/10]
 b←[b/10]
 p←p*10
 -până când a*b=0 sau a%10≠b%10
 a←a*p
 scrie a

d.

```
#include<iostream>
using namespace std;
int main()
{ int a,b,p=1;
cin>>a>>b;
while (a*b>0 && a%10==b%10)
  { a=a/10;
    b=b/10;
    p=p*10;
  }
a=a*p;
cout<<a;
return 0;
}
```

2. float b=0;
 if (v>y)
 b=v*0.10;
 else if (v>x)
 b=v*0.05;
3. int n;
 float t, v;
 cout<<ceil(v/n);

Subiectul III

1. #include<iostream>
 using namespace std;
 int main()
 { int i, ok1, ok2; long a, a1, b, b1, c=0;
 cin>>a>>b;
 for (i=9;i>=1;i-=2)
 { ok1=ok2=0; a1=a; b1=b;
 while (a1>0&&ok1==0) if (a1%10==i)ok1=1;

```

        else a1/=10;
        while (b1>0&&ok2==0)    if (b1%10==i) ok2=1;
                                else b1/=10;
        if (ok1==1 && ok2==1) c=c*10+i;
    }
    cout<<c;
    return 0;
}

2. #include <iostream>
using namespace std;
int main()
{
    int n, m, p, aux, i, j, t[200],jp;
    cin>>m>>p>>n;
    for (int i=0;i<n;i++)
        cin>>t[i];
    j=0; jp=n;
    for (i=0; i<n; i++)
    {
//cele divizibile le punem la final in afara sirului si apoi
//lipim subsirurile
        if (t[i]%p!=0) t[j++]=t[i];
        else t[jp++]=t[i];
    }
    i=j;
    j=n;
    while(i<n)
        t[i++]=t[j++];
    for (i=0;i<n;i++)
    {
        cout<<t[i]<<" ";
    }
    cout<<endl;
    if (jp-n < m)
    {
        cout<<jp-n<<endl;
    }
}

3. #include<iostream>
#include<fstream>
using namespace std;
int v[1000];
ifstream f ("BAC.IN");
int main()
{long x, ok=0;
while (f>>x) if (x<1000)v[x]++;
x=999;
while (x>=100&&ok==0)
    if (v[x]==0 && v[1000-x]!=0) ok=1;
    else x--;
if (ok) cout<<x;
else cout<<"NU EXISTA";
return 0;
}

```

O soluție posibilă folosește un vector de frecvență pentru numerele de maxim 3 cifre care apar în fișier. Apoi se parcurge vectorul de la ultima poziție și se identifica primul număr (poziție) care are frecvența nulă și pentru care se îndeplinește condiția că $v[1000-nr]$ este nenulă. Complexitatea algoritmului este liniară ($O(n)$).

BAREM DE EVALUARE ȘI DE NOTARE

TEST 4

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.

Subiectul I

20 de puncte

1. b	2. d	3. c	4. d	5. c	4p×5
------	------	------	------	------	------

Subiectul II

40 de puncte

4. a. 51400 b. 450 de valori (toate numerele de forma $\overline{xy73}$, $\overline{xy03}$, $\overline{xy53}$, $\overline{xy50}$, $\overline{xy00}$) c. echivalare corectă d. Pentru program corect _declararea corectă a tuturor variabilelor _citire corectă _scriere corectă _instrucțiune repetitivă corectă _atribuiri corecte corectitudinea globală a programului	6p 6p 6p 10p 2p 1p 1p 3p 2p 1p
5. Pentru rezolvarea corectă _declararea corectă a variabilei b _instrucțiunile de test corecte _formula bonusului corectă corectitudine generală	6p 1p 2p 2p 1p
6. Pentru rezolvare corectă _declararea corectă a variabilelor _instrucțiunea de scriere corectă formula de scriere corectă	6p 2p 2p 2p

Subiectul III

30 de puncte

1. Pentru rezolvare corectă _declarații corecte de variabile _citirea corectă a datelor de intrare _determinare cifre distincte comune _determinare număr maxim cerut _afișare corectă corectitudinea globală a programului	10 p 1p 1p 3p 3p 1p 1p
2. Pentru rezolvare corectă _declararea corectă a variabilelor _citirea corectă a datelor de intrare _algoritm principal corect _corectitudinea formală (variabile, structura, sintaxa) afișarea corectă a datelor de ieșire	10p 1p 2p 3p 2p 2p

3. Pentru rezolvare corectă	10p
_ operații cu fișiere (declarare, deschidere)	1p
_ citirea numerelor	1p
_ algoritm principal corect	2p
_ determinarea valorii cerute	2p
_ afișarea rezultatului	1p
_ corectitudine formală (declarare variabile, structură program, sintaxa instrucțiunilor)	1p
_ coerența explicării metodei	1p
_ explicarea unor elemente de eficiență conform cerinței (numai pentru eficiență $O(n)$)	1p
O soluție posibilă folosește un vector de frecvență pentru numerele de maxim 3 cifre care apar în fișier. Apoi se parcurge vectorul de la ultima poziție și se identifică primul număr (poziție) care are frecvența nulă și pentru care se îndeplinește condiția că $v[1000-nr]$ este nenulă. Astfel, algoritmul utilizat este de tip liniar (complexitate $O(n)$).	