


```

citește x (număr natural)
p ← 1; m ← -1
cât timp p ≤ x execută
    c ← [x/p] % 10
    dacă c > m atunci
        m ← c; p ← p * 10
    altfel
        x ← [x / (p * 10)] * p + x % p
dacă m ≥ 0 atunci scrie x
altfel scrie "nul"

```

a. Scrieți valoarea afișată dacă se citește numărul 6907512.

Răspuns:

b. Scrieți două valori distincte din intervalul [100, 999] care pot fi citite astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, să se afișeze o valoare identică cu cea citită.

Răspuns:

6. Completați spațiile lacunare din următoarea secvență de instrucțiuni astfel încât să determine în variabila nr numărul de divizori naturali ai numărului natural nenul x.

```

nr ← 2; d ← 2;
cât timp ... execută
    dacă ... atunci
        nr ← nr + 2;
    d ← d + 1;
dacă ... atunci
    nr ← nr + 1;

```

7. Scrieți un program în limbajul C/C++ sau un algoritm în pseudocod care să citească două numere naturale x și n ($0 < n < 10000$, $0 < x < 10^9$), apoi o succesiune de n numere naturale mai mici decât 10^9 . Programul va afișa numărul de valori din succesiune pentru care x este sufix sau prefix.

Un număr natural x este numit prefix al unui număr natural y dacă se obține din acesta prin eliminarea a cel puțin unei cifre de la dreapta sa, și este numit sufix al lui y dacă se obține din acesta prin eliminarea a cel puțin unei cifre de la stânga sa. De exemplu dacă $x=13$, $n=5$ și succesiunea este 2134, 1213, 1324, 1313, 13, rezultatul va fi 3 (13 este sufix pentru 1213, este prefix pentru 1324 și este atât prefix, cât și sufix pentru 1313).

Barem (10 puncte se acordă din oficiu)

Subiect	1	2	3	4	5a	5b	6	7
Punctaj	10	10	10	10	9	6	15	20

Notă: Baremul detaliat de evaluare și rezultatele evaluării vor fi publicate pe site-ul Centrului de Excelență, disciplina Informatică la adresa ler.is.edu.ro/~cex_is



Barem detaliat de evaluare

Subiect	Punctaj	Soluție	Observații																				
1	10	16	Mașina funcționează astfel: <table border="1"> <thead> <tr> <th>Componenta</th> <th>1</th> <th>2</th> <th>3</th> <th>4</th> </tr> </thead> <tbody> <tr> <td>Nr. 1</td> <td>4</td> <td>3</td> <td>2</td> <td>1</td> </tr> <tr> <td>Nr. 2</td> <td>1</td> <td>3</td> <td>5</td> <td>7</td> </tr> <tr> <td>Nr. 3</td> <td>1</td> <td>4</td> <td>9</td> <td>16</td> </tr> </tbody> </table>	Componenta	1	2	3	4	Nr. 1	4	3	2	1	Nr. 2	1	3	5	7	Nr. 3	1	4	9	16
Componenta	1	2	3	4																			
Nr. 1	4	3	2	1																			
Nr. 2	1	3	5	7																			
Nr. 3	1	4	9	16																			
2	10	29	Valoarea fiecărei monede este egală cu suma numerelor scrise în jurul valorii centrale + numărul de laturi ale monedei: Pătrat: $3 + 5 + 6 + 2 + 4 = 20$ Hexagon: $1 + 9 + 2 + 4 + 5 + 0 + 6 = 27$ Triunghi: $10 + 2 + 3 + 3 = 18$ Pentagon: $5 + 2 + 3 + 8 + 6 + 5 = 29$																				
3	10	c																					
4	10	d																					
5a	9	9752																					
5b	6	oricare două numere distincte din intervalul cerut, cu cifre în ordine strict descrescătoare	Se acordă jumătate din punctaj pentru un singur număr corect și niciunul greșit.																				
6	15	<pre>nr←2; d←2; ┌cât timp d*d<x execută ├─dacă x%d=0 atunci │ ┌nr←nr+2; │ └─ ├─d←d+1; └─ ├─dacă d*d=x atunci │ ┌nr←nr+1; │ └─ └─</pre>	Se acordă 5 puncte pentru fiecare zonă completată corect.																				
7	20	<pre>#include <iostream> using namespace std; int main() {int n, x, a, i, nr=0, p=1, copie; cin>>x>>n; copie=x; while (copie>0) {p*=10; copie/=10;} for (i=0; i<n; i++) {cin>>a; //verific daca x este sufix if (a%p==x && a/p>0) nr++; else //verific daca x e prefix {a/=10; while (a>p) a/=10; if (a==x) nr++; } } cout<<nr; return 0; }</pre>	Se acordă punctaje parțiale: 5p Pentru citirea corectă a datelor 5p Pentru testarea condiției de sufix 5p Pentru testarea condiției de prefix 5p Pentru numărare corectă (fără a număra valorile care sunt atât prefix, cât și sufix de două ori)																				