

citește x, z (numere naturale)

$y \leftarrow 0$

repetă

$y \leftarrow y \cdot 10 + x \% 10$

$x \leftarrow [x/100]$

până când $x=0$

cât timp $y \cdot z > 0$ și $y \% 10 = z \% 10$ execută

$y \leftarrow [y/10]$

$z \leftarrow [z/10]$

■

dacă $y+z=0$ atunci

scrie 1

altfel

scrie 0

■

a) Scrieți ce se va afișa dacă se citesc,
în această ordine, valorile 729385 și
532. (6p.)

var. 5

x	z	y
729385	532	0
7293		5
72		53
		532
	53	53
	\$	\$
	0	0

Răspuns: Scrie 1.

Întrucât această secvență
afișează de fapt 1 dacă $y = z$
sau 0 în caz contrar.

- b) Dacă pentru z se citește valoarea 99, scrieți **câte** numere naturale, cu exact 3 cifre fiecare, pot fi citite pentru x astfel încât să se afișeze valoarea 0 în fiecare dintre aceste cazuri. (4p.)

Fie $x = \overline{abc}$. Dacă $z = 99$, $\Rightarrow x$ tre' să
fie de forma $\overline{9 \neq 9}$: 10 bucăți (909, ... 999)
ca să obținem 1.

Din totalul de 900 (999-100+1) de numere
cu 3 cifre le scădem pe cele 10 la

$$\text{Kre dă } 1 \Rightarrow 900 - 10 = 890$$

de valori pt. care
se obține 0.

citește x, z (numere naturale)

$y \leftarrow 0$

repetă

$y \leftarrow y \cdot 10 + x \% 10$

$x \leftarrow \lfloor x / 100 \rfloor$

până când $x = 0$

cât timp $y \cdot z > 0$ și $y \% 10 = z \% 10$ execută

$y \leftarrow \lfloor y / 10 \rfloor$

$z \leftarrow \lfloor z / 10 \rfloor$

■

c

dacă $y = z$ atunci

 scrie 1

altfel

 scrie 0

■

- c) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se folosească o singură structură repetitivă. (6p.)

rescriem a doua repetitivă ca
după cum am văzut veniă o
condiție elementară

citește x, z

$y \leftarrow 0$

repetă

$y \leftarrow y \cdot 10 + x \% 10$

$x \leftarrow \lfloor x / 100 \rfloor$

până când $x = 0$

dacă $y = z$ atunci

 scrie 1

altfel

 scrie 0

■

```

citește x,z (numere naturale)
y←0
repetă
    y←y*10+x%10
    x←[x/100]
până când x=0
cât timp y*z>0 și y%10=z%10 execută
    y←[y/10]
    z←[z/10]
■
dacă y+z=0 atunci
    scrie 1
altfel
    scrie 0
■

```

d) Scrieți programul c/c++ corespunzător algoritmului dat.

```

#include <iostream>
using namespace std;
int main()
{
    int x,y,z;
    cin>>x>>z;
    y=0;
    do { y = y*10 + x%10;
        x/=100;
    } while (x!=0);
    while (y*z>0 && y%10==z%10)
    {
        y/=10;
        z/=10;
    }
    if (y+z==0)
        cout<<1;
    else
        cout<<0;
    return 0;
}

```

citește n (număr natural) a)

$a \leftarrow n \% 10$

$m \leftarrow a$

cât timp $n > 9$ execută

$n \leftarrow [n/10]$

$b \leftarrow n \% 10$

dacă $a > b$ atunci

$m \leftarrow m * 10 + b$

$a \leftarrow b$

■

■

scrie m

Scrieți valoarea pe care o va afișa algoritmul dacă se citește numărul 16389. (6p.)

Var 20

n	a	b	m
16389	9	8	9
1638	8	3	98
163	3	6	983
16	6	1	9831
1	1		

Răsp: 9831

Algoritmul formează un număr din subșirul descrescător al cifrelor numărului dat. Inițial se alege ultima cifră a numărului dat după care se alege cifra curentă DOAR dacă e mai mică decât ultima cifră aleasă.

- b) Scrieți cea mai mică valoare de patru cifre distincte care poate fi citită pentru variabila n , astfel încât să se afișeze valoarea 0. (4p.)

Răsp: 1230.

```

citește n (număr natural)
a ← n % 10
m ← a
cât timp n > 9 execută
    n ← [n / 10]
    b ← n % 10
    dacă a > b atunci
        m ← m * 10 + b
        a ← b
    ■
scrie m

```

- d) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască structura cât timp...execută cu o structură repetitivă de alt tip.

```

citește n (număr natural)
a ← n % 10
m ← a

```

dacă $n > 9$ execută

```

    execută
        n ← [n / 10]
        b ← n % 10
        dacă a > b atunci
            m ← m * 10 + b
            a ← b
        ■
    cât timp  $n > 9$ 

```

```

scrie m

```

```

citește a, b
  (numere naturale nenule)
c ← 0
repetă
  i ← a % 2
  j ← b % 2
  dacă i + j = 0 atunci
    c ← c + 1
  a ← a * i + (1 - i) * [a / 2]
  b ← b * j + (1 - j) * [b / 2]
până când i * j = 1
scrie c

```

var. 23

Scrieți în pseudocod un algoritm echivalent cu cel dat în care să se înlocuiască structura repetă...până când cu o structură repetitivă cu test inițial. (6p.)

```

citește a, b
c ← 0
i ← a % 2   j ← b % 2
dacă i * j = 0 atunci
  c ← c + 1
  a ← a * i + (1 - i) * [a / 2]
  b ← b * j + (1 - j) * [b / 2]
cât timp i * j ≠ 1 execută
  i ← a % 2   j ← b % 2
  dacă i + j = 0 atunci
    c ← c + 1
  a ← a * i + (1 - i) * [a / 2]
  b ← b * j + (1 - j) * [b / 2]

```

scrie c.

```

citește a, b
  (numere naturale nenule)
c ← 0
repetă
  i ← a % 2
  j ← b % 2
  dacă i + j = 0 atunci
    c ← c + 1
  ■
  a ← a * i + (1 - i) * [a / 2]
  b ← b * j + (1 - j) * [b / 2]
până când i * j = 1
scrie c

```

o variantă ceva
mai scurtă, bonată
pe un artificiu:

citește a, b
 c ← 0
 i ← 0 ; j ← 0

→ artificiu pt. a
asigura inițierea
inițială în repetiție

cât timp $i * j \neq 1$ execută

```

    i ← a % 2
    j ← b % 2
    dacă i + j = 0 atunci
      c ← c + 1
    ■
    a ← a * i + (1 - i) * [a / 2]
    b ← b * j + (1 - j) * [b / 2]
  
```

scrie c

citește n (număr întreg)

dacă $n < 0$ atunci

$n \leftarrow -n$

■

$d \leftarrow 1$

pentru $i \leftarrow 2, [n/2]$ execută

dacă $i | n$ atunci

$d \leftarrow i$

■

■

scrie d

- b) Scrieți în pseudocod un algoritm echivalent cu cel dat în care să se înlocuiască structura pentru...execută cu o structură repetitivă de tip cât timp...execută. (6p.)

citeste n
dacă $n < 0$ atunci
 $n \leftarrow -n$
■
 $d \leftarrow 1$
 $i \leftarrow 2$
cât timp $i \leq [n/2]$ execută
 dacă $i | n$ atunci
 $d \leftarrow i$
 ■
 $i \leftarrow i + 1$
■
scrie d

```

citește n (număr întreg)
dacă n < 0 atunci
    n ← -n
d ← 1
pentru i ← 2, [n/2] execută
    dacă i | n atunci
        d ← i
scrie d

```

Să se rescrie pe execută .. cât timp

```

citește n
dacă n < 0 atunci
    n ← -n
d ← 1
i ← 2
dacă i ≤ [n/2] atunci
    execută
        dacă i | n atunci
            d ← i
        i ← i + 1
    cât timp i ≤ [n/2]
scrie d

```