

```

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

```

a) Scrieți valoarea care se va afișa dacă se citesc, în ordine, valorile 24 și 36. (6p.)

var. 23

a	b	c	i	j
<del>24</del>	<del>36</del>	<del>0</del>	<del>0</del>	<del>0</del>
<del>12</del>	<del>18</del>	<del>1</del>	<del>0</del>	<del>0</del>
<del>6</del>	<del>9</del>	2	<del>0</del>	<del>1</del>
<del>3</del>	<del>3</del>		1	1
3	9			

Răsp.: afișarea 2.

Obs.: Alg. determină exponentul lui 2 din descompunerea în factori primi a numerelor dintre a și b.

- b) Scrieți două valori care trebuie citite (una pentru variabila `a` și una pentru variabila `b`) astfel încât, în urma executării algoritmului, să se afișeze valoarea 0.  
(4p.)

posibil răspuns : 1 1  
este corect orice răspuns format din 2 numere naturale  
din care măcar unul să fie impar.

```

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

```

```

#include <iostream>
using namespace std;
int main()
{ int a,b,i,j,c;
  cin >> a >> b;
  c = 0;
  do { i = a%2;
        j = b%2;
        if ( i+j == 0 )
            c++;
    }

```

```

a = a*i + (1-i)*(a/2);
b = b*j + (1-j)*(b/2);
} while (i*j != 1);
cout << c;
return 0;
}

```

var. 4

citește a, b (numere întregi)

dacă  $a < b$  atunci

$a \leftrightarrow b$

pentru  $x \leftarrow a, b, -1$  execută

dacă  $x \% 2 \neq 0$  atunci

scrie x, ' '

a) Răspuns:

17 15 13 11 9 7 5

Deci afișează în ordine  
descrescătoare nr. întregi  
împere dintre a și b  
(indiferent de ordinea inițială  
a lui a și b)

a) Scrieți ce se afișează pentru  $a=5$  și  $b=17$ .

a	b	z
5	17	17
17	5	16
		15
		14
		13
		12
		11
		10
		9
		8
		7
		6
		5

moment  $\rightarrow$  se afișează

b) Scrieți toate perechile de valori care pot fi citite pentru variabilele a și b, astfel încât să se afișeze, în această ordine, numerele: 1 -1.

răspuns:

-2 1  
-2 2  
-1 1  
-1 2  
1 -2  
1 -1  
2 -2  
2 -1

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

citește a,b (numere întregi)

dacă  $a < b$  atunci

$a \leftrightarrow b$

pentru  $x \leftarrow a, b, -1$  execută

dacă  $x \% 2 \neq 0$  atunci

scrie x, ' '

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

```
#include <iostream>
using namespace std;
```

```
int main()
```

```
{ int a, b, x;
```

```
cin >> a >> b;
```

```
if (a < b) { x = a; a = b; b = x; }
```

```
for (x = a; x >= b; x --)
```

```
if (x % 2) cout << x << " ";
```

```
return 0;
```

```
}
```

citește a,b (numere întregi)

dacă  $a < b$  atunci

$a \leftrightarrow b$

$x \leftarrow a$

cât timp  $x \geq b$  execută

dacă  $x \% 2 \neq 0$  atunci

scrie x, ' '

$x \leftarrow x - 1$

citește  $x, z$  (numere naturale)

$y \leftarrow 0$

repetă

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

$x \leftarrow [x / 100]$

până când  $x = 0$

cât timp  $y * z > 0$  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 05

x	z	y
<del>729385</del>	<del>532</del>	<del>0</del>
<del>7293</del>		<del>8</del>
<del>72</del>		<del>53</del>
0		532
	53	53
	5	5
	0	0

Răsp: scrie 1

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.)

$x = abc$  } se afiș 1  
 $z = 99$  } doar dacă  $\begin{cases} a=9 \\ c=9 \end{cases}$  deci p.  $x = \overline{9 \times 9}$

○ se afișează la rețut de numere cu 3 cifre  
Total există 900 de numere de 3 cifre (100.. 999)  
din acestea sunt 10 bur.  
răsp: 900 - 10 = 890 //

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

```
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
■
```

→ ne putem că, de fapt, secvența  
asta complicată alegea 1 dacă  
și numai dacă  $y == z$ .

Soluție:

```
citește x, z
repetă
    y ← y * 10 + x % 10
    x ← [x / 100]
până când x = 0
dacă y = z atunci
    scrie 1
altfel
    scrie 0
■
```

Ver 27

citeste a, b  
(numere naturale)

$c \leftarrow 0$

$d \leftarrow 0$

$p \leftarrow 1$

cât timp  $a+b+c > 0$  execută

$c \leftarrow a \% 10 + b \% 10 + c$

$d \leftarrow d + (c \% 10) * p$

$p \leftarrow p * 10$

$a \leftarrow [a / 10]$

$b \leftarrow [b / 10]$

$c \leftarrow [c / 10]$

scrie d

a) ce afiș. p1.  $a = 493$  și  $b = 1836$

Răsp: 2 3 2 9

$$\begin{array}{r} 1836 + \\ 493 \\ \hline 2329 \end{array}$$

Deci, de fapt în d se calculează  
suma  $a+b$ .

a	b	c	d	p
<del>493</del>	1836	<del>0</del>	<del>0</del>	<del>1</del>
		9	9	10
<del>49</del>	<del>183</del>	<del>0</del>	29	100
		12		
<del>4</del>	<del>18</del>	<del>1</del>	329	1000
		13		
0	<del>1</del>	<del>1</del>	2329	10000
	0	2		
		0		



```

citeste a,b
           (numere naturale)
c←0
d←0
p←1
cât timp a+b+c>0 execută
|   c←a%10+b%10+c
|   d←d+(c%10)*p
|   p←p*10
|   a←[a/10]
|   b←[b/10]
|   c←[c/10]
■
scrie d

```

d) Scrieți în pseudocod un algoritm echivalent cu cel dat, care să **NU** folosească structuri repetitive sau recursive. (4p.)

*citeste a,b  
scrie a+b*