**Fișa 1. Probleme cu subprograme**

1. **Inversare număr**

Scrieți o funcție cu numele **oglindit**, care să returneze oglinditul unui număr natural **n**, transmis ca parametru.

**Restricții și precizări: n<109**.

1. **Cifra de control**

*Cifra de control* a unui număr se obține efectuând suma cifrelor sale, apoi suma cifrelor acestei sume, până când suma obținută este un număr format dintr-o singură cifră. Această ultimă cifră poartă numele de **cifră de control**.

Scrieți o funcție cu numele **cc**, care primește ca parametru un număr natural **n** și returnează cifra de control a numărului **n**.

***Exemplu*:** Cifra de control a numărului **n=4568248**: **cc(4568258)=2**.

1. **Cifra de control comuna**

Scrieți o funcție cu numele **sum\_digit\_control**, care primește ca argumente două numere naturale **a** și **b** și returnează numărul de elemente din intervalul **[a,b]** care au cifra de control egală cu **a**.

Cifra de control a unui număr se obține efectuând suma cifrelor sale, apoi suma cifrelor acestei sume etc. până se obține o sumă formată dintr-o singură cifră. De exemplu, cifra de control a numărului **645** este **6**. (**6+4+5=15**, **1+5=6**)

**Restricții și precizări**

* funcția va avea doi parametri, **a** și **b**, în această ordine, **1≤a≤9**, **a<b≤107**.

***Exemplu*:** **sum\_digit\_control(2,100)=11**.

1. **Formare număr real**

Scrieți definiția completă a subprogramului **float\_number** cu doi parametri **x** și **y**, numere naturale din intervalul **[1;1000]** ce returnează un număr real cu proprietatea că partea sa întreagă este egală cu **x**, iar numărul format din zecimalele sale, în aceeași ordine, este egal cu **y**.

**Restricții și precizări**

* funcția va avea doi parametri: **x** și **y**, în această ordine;
* rezultatul calculat al funcției va fi de tip **double**.

***Exemplu*:** Pentru **x=12** și **y=543**, subprogramul returnează valoarea **12.543**.

1. **Cifre impare egale**

Subprogramul **OddEqual** are un parametru, **n**, prin care primește un număr natural cu cel puțin o cifră impară. Subprogramul returnează valoarea **1** dacă toate cifrele impare ale lui **n** sunt egale între ele sau valoarea **0** în caz contrar. Scrieți definiția completă a subprogramului.

**Restricții și precizări**

* **10≤n≤109**.

***Exemplu*:** Dacă **n=7727470** sau **n=7240** atunci subprogramul returnează **1**, iar dacă **n=7921470** atunci subprogramul returnează **0**.

1. **Suma divizorilor**

Scrieți o funcție cu numele **suma\_div**, care să determine suma divizorilor unui număr natural **n** transmis ca parametru. Funcția va returna suma în locul de apel.

**Restricții și precizări: n≤2⋅109**.

1. **Numărul maximal\***

Scrieți o funcție cu numele **nrmaxim**, care are ca parametru un număr natural **n** și care returnează *cel mai mare număr* care poate fi obținut mutând, pe rând, prima cifră a numărului **n** și a celor obținute pe parcurs, pe ultima poziție. Nu se vor folosi șiruri de caractere și tablouri auxiliare.

**Restricții și precizări: n≤109**.

***Exemplu*:** **nrmaxim(n=4273)=7342** – este cel mai mare număr dintre **2734**, **7342**, **3427**, **4273**.

1. **Suma factorialelor cifrelor\***

Scrieți o funcție cu numele **sumafactcifre**, care să returneze suma factorialelor cifrelor unui număr natural **n**, transmis ca parametru funcției.

**Restricții și precizări**

* funcția va avea un parametru reprezentând numărul dat, care va fi mai mic decât **2.000.000.000**;
* prin definiție, **0!=1**.

***Exemplu***: **sumafactcifre(241)=27**, adică **2!+4!+1!=1\*2+1\*2\*3\*4+1=2+24+1=27**.

1. **P-numere\***

Un număr natural nenul se numește **p-număr** dacă are aceeași *paritate cu suma divizorilor săi pozitivi*.

***Exemplu*:** **10** și **25** sunt **p-numere** (**10** are aceeași paritate cu **18=1+2+5+10**, iar **25** are aceeași paritate cu **31=1+5+25**).

Scrieți un subprogram cu numele **kpn**, are trei parametri, **a**, **b** și **k**, în această ordine, prin care primește câte un număr natural din intervalul **[1,105]** (**a≤b**). Subprogramul returnează cel de al **k**-lea **p**-număr din intervalul **[a,b]** sau **-1**, dacă nu există cel puțin **k** astfel de numere în acest interval. Scrieți definiția completă a subprogramului.

***Exemplu*:** Dacă **a=27**, **b=50** și **k=3**, atunci subprogramul returnează **34**.

1. **Diferența minimă a două numere prime\***

Scrieți definiția completă a unui subprogram cu numele **i\_prim** care primește prin singurul său parametru, **n**, un număr natural cu cel mult **9** cifre și returnează diferența minimă **p2-p1** în care **p1** și **p2** sunt numere prime și **p1≤n≤p2**.

**Restricții și precizări:** **n>2**.

**Exemplu:** Dacă **n=28**, **i\_prim(n)=6**, deoarece **p1=23** și **p2=29**.

1. **Numărare numere naturale nenule, prime cu n\***

Scrieți funcția cu numele **Count\_prim\_n**, care primește ca parametru un număr natural **n** și care trebuie să returneze *numărul de numere naturale nenule prime cu* **n** și strict mai mici decât **n**.

**Restricții și precizări**

* **2≤n≤2.000.000.000**.
* Numele funcției este Phi.

Spunem că un număr natural **x** este prim cu **n**, dacă **cmmdc(x,n)=1**.

***Exemplu*:** **Count\_prim\_n(12)=4**, deoarece **12** este prim cu numerele **1**, **5**, **7**, **11**.

1. **Număr factorial aproape de n\***

Scrieți o funcție cu numele **proxim**, cu un parametru, **n**, care returnează *cel mai apropiat număr de* **n** *care este factorialul unei valori*.

**Restricții și precizări**

* **0≤n≤10.000.000**
* dacă există mai multe numere care sunt factoriale ale unor valori și sunt la fel de apropiate de **n**, funcția îl va returna pe cel mai mic.

**Exemplu: proxim(24)=24**, **proxim(44)=24**, **proxim(100)=120**, **proxim(72)=24**.

1. **Numere norocoase\***

Se spune despre numărul natural nenul **n** că este **norocos** dacă numărul **n2** se poate scrie ca suma a **n** numere naturale nenule consecutive. De exemplu, **7** este norocos, deoarece **72=4+5+6+7+8+9+10**.

Scrieți definiția completă a funcției **norocoase**, care primește ca argumente două numere naturale **a** și **b** și returnează câte numere norocoase se află în intervalul **[a,b]**.

**Restricții și precizări:**

* **1≤a<b≤109**.
* numele funcției este **norocoase**, parametrii sunt, în această ordine: **a**, **b**.

***Exemplu*:** **norocoase(1,15)=8**.

**Notă**. Puteți defini și alte subprograme care vă ajută să rezolvați mai ușor problema.

1. **Număr de zerouri la sfârșitul lui n!\***

Scrieți definiția completă a unui subprogram cu numele **nrzero**, cu un parametru întreg **n**, care returnează numărul zerourilor de la sfârșitul numărului **n!**.

**Restricții și precizări: 0<n<109**.

**Exemplu:** Dacă **n=12**, **nrzero(n)=2**, deoarece **12!** se termină cu două cifre zero.

1. **Poziție reciproca dreptunghiuri\***

Considerăm un sistemul cartezian de coordonate **xOy** astfel:

* colțul stânga-sus are coordonata (**0**, **0**)
* axa **Ox** începe din stânga și continuă crescător spre dreapta
* axa **Oy** începe de sus și continuă crescător în jos

În imaginea de mai jos sunt afișate coordonatele unui punct în sistemul cartezian folosit:

0

*x*

*P*(*x*,*y*)

*y*

În acest sistem de coordonate, un dreptunghi cu laturile paralele cu axele poate fi descris prin patru numere, reprezentând:

* **x** – coordonata **X** a punctului din stânga – sus
* **y** – coordonata **Y** a punctului din stânga – sus
* **l** – lățimea dreptunghiului
* **i** – înălțimea dreptunghiului,

Scrieți definiția funcției **poz\_reciproca** care, primind **8** argumente, **x1**, **y1**, **l1**, **i1**, **x2**, **y2**, **l2**, **i2**, în această ordine:

* **x1**, **y1**, **l1**, **i1** – reprezentând primul dreptunghi,
* **x2**, **y2**, **l2**, **i2** – reprezentând cel de-al doilea dreptunghi,

returnează:

* **0**, dacă primul dreptunghi este în interiorul celuilalt sau invers
* **1**, dacă cele două dreptunghiuri sunt identice
* **2**, dacă cele două dreptunghiuri nu au nici un punct comun
* **3**, dacă cele două dreptunghiuri se suprapun doar parțial

**Restricții și precizări**

* specificațiile unui dreptunghi sunt numere naturale cu cel mult **4** cifre fiecare;
* soluția va conține funcția cerută. Prezența în soluție a altor instrucțiuni poate duce la erori de compilare și depunctarea soluției

***Exemple*:**

**poz(100, 100, 300, 200, 150, 150, 100, 130)** returnează **0**

**poz(100, 100, 100, 200, 100, 100, 100, 200)** returnează **1**

**poz(50, 50, 100, 50, 200, 100, 100, 130)** returnează **2**

**poz(50, 50, 100, 50, 30, 40, 100, 130)** returnează **3**

**Cu tablouri**

1. **Cel mai mare/cel mai mic număr din cifrele lui n\***

Scrieți câte o funcție care să returneze *cel mai mare*, respectiv *cel mai mic* număr care se poate scrie cu cifrele unui număr natural **n**, transmis ca parametru.

**Restricții și precizări**

* numele funcțiilor va fi **maxnr**, **minnr**;
* funcția va avea un parametru **n**, reprezentând numărul care se dă;
* **n≤109**.