1.Scrieți un program care citește un număr natural n și care să calculeze și să afișeze suma S a tuturor numerelor obținute prin rearanjarea cifrelor lui n.

#**include** <iostream>

**using** **namespace** std;

**int** **main**() {

**int** n, sc = 0, nc = 0, x = 0, f = 1;

cin >> n;

**if** (n == 0) nc =1;

**while** (n > 0) {

x = x \* 10 + 1;

nc++;

f = f \* nc;

sc = sc + n % 10;

n = n / 10;

}

f = f / nc;

cout << 1LL \* f \* x \* sc;

**return** 0; }

2.Se dă un număr natural n. Să se afle câte dintre numerele obținute din n prin ștergerea unei cifre, sunt divizibile cu 3.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** main()

{

**int** n,x,cate=0;

cin>>n;

x=n%3;

**while**(n>0)

{

**if**((n%10)%3==x) cate++;

n=n/10;

}

cout<<cate;

**return** 0; }

3.Se dă n număr natural. Aflaţi cel mai mare număr natural care are numărul de cifre şi suma cifrelor egale cu suma cifrelor lui n.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** main()

{

**int** n,aux,s=0,p=0,k=0;

cin>>n;

**while**(n!=0)

{

s=s+n%10;

n/=10;

}

aux=s;

**while**(s>9)

{

cout<<"9";

k++;

s=s-9;

}

cout<<s;

**while**(p<aux-k-1)

{

cout<<"0";

p++;

}

4. Se dau n numere naturale. Se numește număr par asociat unui număr x numărul obținut din cifrele pare ale lui x luate în ordinea în care apar ele în x. Se cere să se determine câte dintre cele n numere citite au numărul par asociat palindrom.

**int** main(){

**int** n, cnt=0, x;

cin >> n;

**for**(**int** i = 1; i <= n; ++i){

cin >> x;

**int** nr = 0, put = 1;

**bool** ok = **false**; /// o sa stiu daca numarul a continut cifre pare

**if**(x == 0)

cnt++;

**while**(x){ /// calculez numerele pare

**if**(x % 2 == 0)

ok = **true**, nr = (x % 10) \* put + nr, put \*= 10;

x/=10;

}

**if**(ok){

**int** a = nr, b = 0;

**while**(nr){ /// calculez rasturnatul numarului par

b = b \* 10 + nr % 10;

nr/=10;

}

**if**(a == b) /// compar rasturnatele sa vad daca este palindrom

cnt++; /// maresc un contor ca sa tin minte cate sunt

}

}

cout << cnt;

**return** 0;}

5.Se citesc n numere naturale, să se pe afișeze pe câte o linie, fiecare pereche de numere consecutive care au ultima cifră egală cu prima cifra a numărului urmator.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** primacifra(**int** n)

{

**while**(n > 9)

n/=10;

**return** n;

}

**int** main()

{

**int** n;

cin >> n;

**int** x , y;

cin >> x;

**for**(**int** i = 2 ; i <= n ; ++i)

{

cin >> y;

**if**(x % 10 == primacifra(y))

cout << x << ' ' << y << '\n';

x=y;

}

**return** 0;

}

6. Se dă un număr natural n. Să se determine numărul maxim care se poate obține din n eliminând exact o cifră. Cifrele rămase nu-și pot schimba ordinea.

???

7.Se citesc perechi de numere naturale până la citirea a două valori nule. Să se determine câte dintre perechi încep cu aceeași cifră.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** andr(**int** x)

{

**while** (x > 9)

{

x=x/10;

}

**return** x;

}

**int** main()

{

**int** a , b , cnt = 0;

cin >> a >> b;

**while**(a != 0 && b != 0)

{

**if**(andr(a)==andr(b)) cnt++;

cin >> a >> b;

}

cout << cnt;

}

8.Se citesc de la tastatură n numere naturale. Să se determine numărul a cărui sumă a cifrelor este cea mai mare, respective cea mai mică.

**int** main ()

{

**int** n,tmp,s=0;

**int** max=0,min=100,smin=100,smax=0;

**int** a;

cin >> n;

**for** (**int** i=1; i <= n; ++i)

{

cin >> tmp;

a=tmp;

**while** (tmp)

{

s+=tmp%10;

tmp/=10;

}

**if** (s>smax)

smax = s ,max = a;

**if** (s<smin)

smin = s ,min = a;

s=0;

}

cout << min << endl << max;

**return** 0;

}

9.Se dau mai multe numere naturale. Determinaţi cel mai mare număr palindrom aflat printre numerele date şi de câte ori apare.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** pal(**int** n)

{

**int** ogl = 0 , aux = n;

**while**(n!=0)

{

ogl = ogl\*10+n%10;

n/=10;

}

**if**(aux == ogl) **return** 1;

**else** **return** 0;

}

**int** main()

{

**int** n,aux,maxim=-1,cnt=0;

cin>>n;

**while**(n!=0)

{

**if**(pal(n) && n > maxim) {maxim = n; cnt=1;}

**else** **if**(pal(n) && n == maxim) cnt++;

cin>>n;

}

**if**(maxim==-1) cout<<"NU EXISTA";

**else**

cout<<maxim<<" "<<cnt;

**return** 0; }

10.Scrieți un program care citește un număr natural n și care să calculeze suma S a tuturor numerelor obținute prin permutări circulare la dreapta ale cifrelor lui n cu o poziție.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** main()

{

**int** n;

cin >> n;

**int** s = 0;

**int** cnt = 0;

**while**(n)

{

s+=n%10;

n/=10;

cnt++;

}

**long** **long** **int** nr = 0;

**int** tr = 0;

**int** a[20];

**for**(**int** i = cnt; i > 0; --i)

a[i] = (s + tr) % 10, tr = (s+tr) / 10;

**if**(tr != 0)

cout << tr;

**for**(**int** i = 1; i <= cnt; ++i)

cout << a[i];

**return** 0; }

11.Se dau n numere naturale. Determinați numărul pentru care prima cifră este maximă.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** main()

{

**int** n,x,aux,care,u,maxim=0;

cin>>n;

**for**(**int** i=1;i<=n;i++)

{

cin>>x;

aux=x;

**int** ogl=0;

**while**(x!=0)

{

ogl=ogl\*10+x%10;

x=x/10;

}

u=ogl%10;

**if**(u>maxim)

{

maxim=u;

care=aux;

}

**else**

**if**(u==maxim)

{

**if**(aux>care)

care=aux;

**else** care=care;

}

}

cout<<care;

**return** 0; }

12.Se citesc două numere naturale. Să se afișeze cel mai mic și cel mai mare număr format din exact două cifre, scris cu o cifră din primul număr și cu o cifră din cel de-al doilea.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**long** **long** x , y , a[20] , b[20] , mini , maxi , c1 , c2;

**int** main()

{

cin >> x >> y;

mini = 100;

**while**(x)

{

a[++c1] = x % 10;

x /= 10;

}

**while**(y)

{

b[++c2] = y % 10;

y /= 10;

}

**for**(**int** i = 1 ; i <= c1 ; i++)

{

**for**(**int** j = 1 ; j <= c2 ; j++)

{

**if**(a[i] \* 10 + b[j] > 9)

{

mini = min(mini , a[i] \* 10 + b[j]);

maxi = max(maxi , a[i] \* 10 + b[j]);

}

**if**(b[j] \* 10 + a[i] > 9)

{

mini = min(mini , b[j] \* 10 + a[i]);

maxi = max(maxi , b[j] \* 10 + a[i]);

}

}

}

cout << mini << " " << maxi;

**return** 0; }

13.Se dau n numere naturale distincte. Pentru oricare două numere date, distincte, x şi y, se determină ultima cifră a numărului 333x•y şi se însumează cifrele obţinute. Aflaţi valoarea acestei sume.

**#include <bits/stdc++.h>**

**using** **namespace** std;

ifstream cin("3lan.in");

ofstream cout("3lan.out");

**int** n , a[1001] , s;

**int** treilan(**int** x , **int** y)

{

**int** rest = (x \* y) % 4 , nr1 , nr2;

**if**(rest == 0) nr1 = 1;

**else** **if**(rest == 1) nr1 = 3;

**else** **if**(rest == 2) nr1 = 9;

**else** **if**(rest == 3) nr1 = 7;

nr1 %= 4;

**if**(nr1 == 0) nr2 = 1;

**else** **if**(nr1 == 1) nr2 = 3;

**else** **if**(nr1 == 2) nr2 = 9;

**else** **if**(nr1 == 3) nr2 = 7;

nr2 %= 4;

**if**(nr2 == 0) **return** 1;

**else** **if**(nr2 == 1) **return** 3;

**else** **if**(nr2 == 2) **return** 9;

**else** **if**(nr2 == 3) **return** 7;

}

**int** main()

{

cin >> n;

**for**(**int** i = 1 ; i <= n ; i++)

cin >> a[i];

**for**(**int** i = 1 ; i <= n ; i++)

**for**(**int** j = i + 1 ; j <= n ; j++)

s += treilan(i , j);

cout << s;

**return** 0; }

14.Se dau n întrebări de forma: **Câte palindromuri există în intervalul [a,b]?**, unde a și b sunt numere naturale date, cu a ≤ b.

**#include <bits/stdc++.h>**

**using** **namespace** std;

ifstream cin("nr\_pal.in");

ofstream cout("nr\_pal.out");

**int** nrcif(**int** n)

{

**int** cnt = 0;

**if**(n <= 9)

**return** 1;

**while**(n)

{

cnt++;

n /= 10;

}

**return** cnt;

}

**int** put(**int** n)

{

**int** p = 1;

**for**(**int** i = 1 ; i <= n ; i++) p \*= 10;

**return** p;

}

**int** f(**int** n)

{

**int** ogl = n;

**while**(n)

{

ogl = ogl \* 10 + n % 10;

n /= 10;

}

**return** ogl;

}

**int** ff(**int** n)

{

**int** ogl = n;

n /= 10;

**while**(n)

{

ogl = ogl \* 10 + n % 10;

n /= 10;

}

**return** ogl;

}

**int** pal(**int** n)

{

**int** cnt = 0;

**int** nr = nrcif(n);

**for**(**int** i = 1 ; i < nr ; ++i)

cnt = cnt+ put((i - 1) / 2) \* 9;

**int** p = put((nr - 1)/2);

**int** y = n/put((nr / 2));

cnt = cnt + (y - p + 1);

**if**(nr % 2 == 0)

{

**if**(f(y) > n) cnt--;

}

**else** **if**(ff(y) > n) cnt--;

**return** cnt;

}

**int** main()

{

**int** n;

**int** a , b;

cin >> n;

**for**(**int** i = 0 ; i < n ; ++i)

{

cin >> a >> b;

cout << pal(b) - pal(a - 1) << '\n';

}

**return** 0; }

15.Se dau două numere naturale a b. Determinați câte numere naturale din intervalul [a,b] se divid cu toate cifrele lor nenule.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** main()

{

**long** **long** a,b,aux,x,k=0,cate,i,p=0,y;

cin>>a>>b;

**for**(**int** i=a;i<=b;i++)

{

x=i;

aux=i;

y=i;

cate=0;

p=0;

**while**(x!=0)

{

**if**(x%10!=0)

{

**if**(aux%(x%10)==0) cate++;

}

p++;

**if**(x%10==0) cate++;

x=x/10;

}

**if**(p==cate) k++;

}

cout<<k;

**return** 0; }

16.Se dau 2 numere naturale a b, a < b. Determinați câte numere din intervalul [a,b] sunt pătrate perfecte și au proprietatea că oglinditul lor este pătrat perfect.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** patrat(**int** n)

{

**if** (sqrt(n)==(**int**)(sqrt(n))) **return** n;

**return** 0;

}

**int** oglindit(**int** n)

{

**int** ogl=0;

**while**(n!=0)

{

ogl=ogl\*10+n%10;

n/=10;

}

**return** ogl;

}

**int** main()

{

**int** a,b,cate=0;

cin>>a>>b;

**for**(**int** i=sqrt(a); i \* i <= b;i++)

{

**if** (i \* i >= a)

{

**int** aux=i \* i;

**if**(patrat(aux))

{

**int** ogl=oglindit(aux);

**if**(patrat(ogl)) cate++;}

}

}

cout<<cate;

**return** 0; }

17.Se citește un număr natural n. Acest număr se “împarte” în alte două numere a și b, astfel: a este format din cifrele din prima jumătate a lui n, b este format din cifrele din a doua jumătate a lui n. Dacă n are număr impar de cifre, cifra din mijloc se ignoră. De exemplu, dacă n=9183792, atunci a=918, iar b=792. Să se determine valoarea absolută a diferenței dintre a și b.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** main ()

{

**int** n;

cin >> n;

**int** m=n;

**int** a , b;

**int** cnt=0;

**int** p = 1;

**int** r;

**while**(n)

{

n/=10;

cnt++;

}

**for** (**int** i = 1; i <= cnt/2; ++i)

p\*=10;

**if** (cnt%2==0)

{

b = m % p;

a = m / p;

}

**else** **if** (cnt%2==1)

{

b = m % p;

a = m / (p\*10);

}

**while** (b)

{

r=a%b;

a=b;

b=r;

}

cout << a;

**return** 0; }

18.Se citesc perechi de numere naturale până la citirea a două valori nule. Să se determine câte dintre perechile X Y au proprietatea că prin concatenarea lui X cu Y sau a lui Y cu X să se obțină un palindrom.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** main()

{

**int** x,y,cate=0;

cin>>x>>y;

**while**(x!=0 || y!=0)

{

///calculez puterea lui 10 < x

**int** px=1;

**while**(x>=px) px=px\*10;

///calculez puterea lui 10 < y

**int** py=1;

**while**(y>=py) py=py\*10;

///calculez cocatenatele

unsigned **long** **long** xy,yx;

xy=1ull\*x\*py+y;

yx=1ull\*y\*px+x;

///calculez resturnatele

unsigned **long** **long** rxy=0,ryx=0;

unsigned **long** **long** cxy=xy,cyx=yx;

**while**(cxy>0)

{

rxy=rxy\*10+cxy%10;

cxy=cxy/10;

}

**while**(cyx>0)

{

ryx=ryx\*10+cyx%10;

cyx=cyx/10;

}

///verific daca e palindrom

**if**(xy==rxy || yx==ryx) cate++;

cin>>x>>y;

}

cout<<cate;

**return** 0;

19.Se citesc n numere naturale. Determinați câte perechi de numere citite consecutiv au aceeași sumă a cifrelor.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** main()

{

**long** **long** n , cnt=0,x;

cin>>n>>x;

**for** (**int** i=1;i<n;i++)

{

**int** pr=x;

cin>>x;

**int** aux=x;

**int** a=pr;

**int** s1=0,s2=0;

**while** (aux!=0)

{

s1=s1+aux%10;

aux=aux/10;

}

**while** (a!=0)

{

s2=s2+a%10;

a=a/10;

}

**if**(s1==s2) cnt++;

}

cout<<cnt;

**return** 0;

20.Se citește o cifră k și apoi se citesc numere până la apariția lui 0. Să se determine de câte ori apare cifra k în numerele citite care sunt pare.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** main()

{

**int** n,k,x,cate=0;

cin >> k >> n;

**while**(n != 0)

{

**if**(n%2==0)

{

**while**(n != 0)

{

x=n%10;

**if**(x==k) cate++;

n=n/10;

}

}

cin >> n;

}

cout << cate;

**return** 0; }

21.Se citesc n numere naturale. Determinați în câte perechi citite consecutiv numerele au sumele cifrelor de parități diferite.

**#include <bits/stdc++.h>**

**using** **namespace** std;

**int** main()

{

**long** **long** n , cnt=0,x;

cin>>n>>x;

**for** (**int** i=1;i<n;i++)

{

**int** pr=x;

cin>>x;

**int** aux=x;

**int** a=pr;

**int** s1=0,s2=0;

**while** (aux!=0)

{

s1=s1+aux%10;

aux=aux/10;

}

**while** (a!=0)

{

s2=s2+a%10;

a=a/10;

}

**if** (s1%2==0 && s2%2==1) cnt++;

**if** (s1%2==1 && s2%2==0) cnt++;

}

cout<<cnt;

**return** 0; }