**Задача A**

n = int(input())

x = []

y = []

**for** i **in** range(n):

cx, cy = map(int, input().split())

x.append(cx)

y.append(cy)

x.sort()

y.sort()

ox = x[len(x) // 2] - x[(len(x) - 1) // 2] + 1

oy = y[len(y) // 2] - y[(len(y) - 1) // 2] + 1

print(ox \* oy)

**Задача B**

**import** math

m = int(input())

otvet = math.ceil(m / 5)

print(otvet)

**Задача C**

#include <bits/stdc++.h>

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

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

{

**int** n, zz;

**cin** >> n;

**vector**<**int**> a;

**vector**<**int**> b;

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

{

**cin** >> zz;

a.push\_back(zz);

}

sort(a.begin(),a.end());

**if** (n == 2)

{

**cout** << a[1] + a[0];

**return** 0;

}

b.push\_back(a[1] - a[0]);

b.push\_back(a[2] - a[0]);

**for**(**int** i = 2; i < a.size() - 1; ++i)

b.push\_back(min(b[i-1],b[i-2]) + **abs**(a[i] - a[i+1]));

**cout** << b[b.size() - 1];

**return** 0;

}

**Задача D**

**Задача E**

n = int(input())

m = int(input())

a = []

c = []

otvet = ""

**for** i **in** range(len(str(n))):

a.append(int(str(n)[i]))

**for** j **in** range(m):

b = a.index(max(a))

a[b] = 0

c.append(b + 1)

c.sort()

**for** h **in** range(len(c)):

otvet += str(c[h]) + " "

print(otvet)

**Задача F**

Program zadachka;

const n = 100;

type lis = array [1..n] of integer;

var q, w, e, r, t, y: integer; arr:lis; z,zz:boolean;

Begin

readln(q);

w:=1;

y:=q;

while y < 10 do

begin

y:=y \* q;

w:=w + 1;

end;

z:=1=1;

r:=2;

arr[1]:=y mod 100;

arr[2]:=arr[1]\*q mod 100;

while (arr[r-1]<>arr[r]) and z do

begin

t:=1;

zz:=1=1;

r:=r+1;

y:=(arr[r-1]\*q) mod 100;

while (t<r) and zz do

if y<>arr[t] then t:=t+1

else zz:=1=2;

if zz then

arr[r]:=y

else z:=1=2;

end;

if w = 1 then

e:=r

else

e:=w+r-1;

write(w, ' ', e);

end.

**Задача G**

n = str(input())

sl = ""

bukvi = {}

un = 0

**for** c **in** n:

**if** c **not** **in** sl:

bukvi[c] = n.count(c)

sl += c

**if** bukvi[c] == 1:

un += 1

print(un)

**Задача H**

x, y = map(int, input().split())

ma = min(x, y)

mi = int(max(x, y)) // 2

otvet = str(mi) + ' ' + str(ma)

print(otvet)