Paranidharan.R

240801238

ECE D

Questiontext1

Youaregivenatwo-dimensional3*3arraystartingfromA[0][0]. Youshouldaddthealternateelementsofthearray andprintitssum.Itshouldprinttwodifferentnumbersthefirstbeing sum of A 0 0, A 0 2, A 1 1, A 2 0, A 2 2 and A 0 1, A10,A12,A21.

InputFormat

Firstandonlylinecontainsthevalueofarrayseparatedbysingle space.

OutputFormat

FirstlineshouldprintsumofA00,A02,A11,A20,A22
SecondlineshouldprintsumofA01,A10,A12,A21

SAMPLEINPUT

123456789

SAMPLEOUTPUT

25
20

```
 1  #include<stdio.h>
 2  int main(){
 3      int arr[3][3];
 4      /*int s1=0,s2=0;*/
 5      for(int i=0;i<3;i++){
 6          for(int j=0;j<3;j++){
 7              scanf("%d",&arr[i][j]);
 8          }
 9      }
10      int odd=0,even=0;
11      for(int i=0;i<3;i++)
12      {
13          for(int j=0;j<3;j++)
14          {
15              if((i+j)%2!=0)
16              odd+=arr[i][j];
17              else
18              even+=arr[i][j];
19          }
20      }
21      printf("%d\n%d",even,odd);
22  }
```

	Input	Expected	Got	
✓	1 2 3 4 5 6 7 8 9	25 20	25 20	✓
✓	21 422 423 443 586 645 657 846 904	2591 2356	2591 2356	✓

Passed all tests! ✓

Questiontext2

Microsofthascometohireinternsfromyourcollege.Nstudentsgot shortlisted out of which few were males and a few females.Allthestudentshavebeenassignedtalentlevels.Smallerthe talent level, lesser is your chance to be selected. Microsoftwantstocreatetheresultlistwhereitwantsthecandidates sorted according to their talent levels, but there isacatch.ThistimeMicrosoftwantstohirefemalecandidatesfirst and then male candidates. Thetaskistocreatealistwherefirstall-femalecandidatesaresorted in a descending order and then male candidates aresortedinadescendingorder. Input Format ThefirstlinecontainsanintegerNdenotingthenumberofstudents. Next,Nlinescontaintwospace-separated integers, ai and bi. Thefirstinteger,aiwillbeeither1(foramalecandidate)or0(for female candidate).

Thesecondinteger,biwillbethecandidate'stalentlevel. Constraints

1<=N<=105

0<=ai<=1

1<=bi<=109

OutputFormat

Outputspace-separatedintegers,whichfirstcontainsthetalentlevels of all female candidates sorted in descending orderandthenthetalentlevelsofmalecandidatesindescending order.

SAMPLEINPUT

5

03

16

02

07

115

SAMPLEOUTPUT

7 3 2 15 6

```c
#include<stdio.h>
struct data
{
    int gen;
    int tal;

};
int main()
{
    int n;
    scanf("%d",&n);
    struct data a[n];
    for(int i=0;i<n;i++)
    scanf("%d %d",&a[i].gen,&a[i].tal);
    for(int i=0;i<n-1;i++)
    {
        for(int j=0;j<n-i-1;++j)
        {
            if(a[j].tal<a[j+1].tal)
            {
                struct data temp=a[j];
                a[j]=a[j+1];
                a[j+1]=temp;

            }
        }
    }
    for(int i=0;i<n;i++)
    {
        if(a[i].gen==0)
        printf("%d ",a[i].tal);
    }for(int i=0;i<n;++i)
    {
        if(a[i].gen==1)
        printf("%d ",a[i].tal);
    }
}
```

	Input	Expected	Got	
✓	5 0 3 1 6 0 2 0 7 1 15	7 3 2 15 6	7 3 2 15 6	✓
✓	6 0 1 0 26 0 39 0 37 0 7 0 13	39 37 26 13 7 1	39 37 26 13 7 1	✓
✓	12 1 12 1 14 1 18 1 1 1 2 1 3 1 5 1 8 1 9 1 10 0 29 0 31	31 29 18 14 12 10 9 8 5 3 2 1	31 29 18 14 12 10 9 8 5 3 2 1	✓
✓	12 0 12 1 12 0 12 1 12 0 12 0 12 1 12 0 12 1 12 1 12 0 12 1 12	12 12 12 12 12 12 12 12 12 12 12 12	12 12 12 12 12 12 12 12 12 12 12 12	✓

Questiontext3

ShyamLal,awealthylandlordfromthestateofRajasthan,beingan old fellow and tired of doing hard work, decided to sellallhisfarmlandandtoliverestofhislifewiththatmoney.No other farmer is rich enough to buy all his land so he decidedtopartitionthelandintorectangularplotsofdifferentsizes with different cost per unit area. So, he sold these plotstothefarmersbutmadeamistake.Beingilliterate,hemade partitions that could be overlapping. When the farmerscametoknowaboutit,theyrantohimforcompensationof extra money they paid to him. So, he decided to returnallthemoneytothefarmersofthatlandwhichwas overlapping with other farmer's land to settle down the conflict.Alltheportionofconflictedlandwillbetakenbackbythe landlord.

To decide the total compensation, he has to calculate the total amount of money to return back to farmers with the same cost they had purchased from him. Suppose, Shyam Lal has a total land area of *1000 x 1000* equal square blocks where each block is equivalent to a unit square area which can be represented on the co-ordinate axis. Now find the total amount of money, he has to return to the farmers. Help Shyam Lal to accomplish this task.

InputFormat:

The first line of the input contains an integer *N*, denoting the total number of land pieces he had distributed. Next *N* line contains the *5* space separated integers *(X1,Y1)*,*(X2,Y2)* to represent a rectangular piece of land, and cost per unit area *C*.

(X1,Y1) and *(X2,Y2)* are the locations of first and last square block on the diagonal of the rectangular region.

OutputFormat:

Print the total amount he has to return to farmers to solve the conflict.

Constraints:

1≤N≤100

1≤X1≤X2≤1000

1≤Y1≤Y2≤1000

1 ≤ C ≤ 1000

SAMPLEINPUT

3

14461

43662

22543

SAMPLEOUTPUT

35

Explanation

87vk

For given sample input (see given graph for reference), compensation money for different farmers is as follows:

Farmer with land area A: C1 = 5 * 1 = 5

Farmer with land area B: C2 = 6*2 = 12

Farmer with land area C: C3 = 6*3 = 18

Total Compensation Money = C1+C2+C3 = 5+12+18 = 35

```c
#include<stdio.h>
int main()
{
    int i,j,n,x1,x2,y1,y2,t=0;
    long long total=0;
    int arr[1001][1001]={0};
    scanf("%d",&n);
    while(n--)
    {
        scanf("%d %d %d %d %d",&x1,&y1,&x2,&y2,&t);
        for(i=x1;i<=x2;i++)
        {
            for(j=y1;j<=y2;j++)
            {
                if(arr[i][j]==0)
                arr[i][j]+=t;
                else if(arr[i][j]>0)
                arr[i][j]=(-1)*(arr[i][j]+t);
                else if(arr[i][j]<0)
                arr[i][j]-=t;
            }
        }
    }
    for(i=1;i<1001;i++)
    {
        for(j=1;j<1001;j++)
        {
            if(arr[i][j]<0)
            total+=arr[i][j];
        }
    }
    printf("%lld\n",(-1)*total);
}
```

	Input	Expected	Got	
✓	3 1 4 4 6 1 4 3 6 6 2 2 2 5 4 3	35	35	✓
✓	1 48 12 49 27 8	0	0	✓
✓	3 88 34 99 76 44 82 65 94 100 81 58 16 65 66 7	10500	10500	✓

Passed all tests! ✓