

Answer: (penalty regime: 0 %)

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

	Expected	Got	
4 5 6 7 8 9	25 20	25 20	✓
423 443 586 645 657 846 904	2591 2356	2591 2356	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 struct data{
3     int gen;
4     int tal;};
5
6 int main(){
7     int n,i,j;
8     scanf("%d",&n);
9     struct data a[n];
10    for(i=0;i<n;i++)
11        scanf("%d %d",&a[i].gen,&a[i].tal);
12
13    for(i=0;i<n-1;i++){
14        for(j=0;j<n-1;j++){
15            if(a[j].tal<a[j+1].tal){
16                struct data temp=a[j];
17                a[j]=a[j+1];
18                a[j+1]=temp;
19            }
20        }
21    }
22    for(i=0;i<n;i++){
23        if(a[i].gen==0)
24            printf("%d ",a[i].tal);}
25    for (i=0;i<n;i++){
26        if(a[i].gen==1)
27            printf("%d ",a[i].tal);
28    }
29
30 }
```

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

Passed all tests! ✓


```
1 #include<stdio.h>
2 int main(){
3     int i,j,n,x1,x2,y1,y2,t=0;
4     long long total=0;
5     int arr[1001][1001]={0};
6     scanf("%d",&n);
7     while(n--){
8         scanf("%d %d %d %d %d",&x1,&y1,&x2,&y2,&t);
9         for(i=x1;i<=x2;i++){
10            for(j=y1;j<=y2;j++){
11                if(arr[i][j]==0)arr[i][j]=t;
12                else if(arr[i][j]>0)arr[i][j]=max(arr[i][j],t);
13                else if(arr[i][j]<0)arr[i][j]=min(arr[i][j],t);
14            }
15        }
16    }
17    for(i=1;i<1001;i++){
18        for(j=1;j<1001;j++){
19            if(arr[i][j]<0)total+=arr[i][j];
20        }
21    }
22    printf("%lld\n",(-1)*total);
23    return 0;
24 }
25
```

	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! ✓