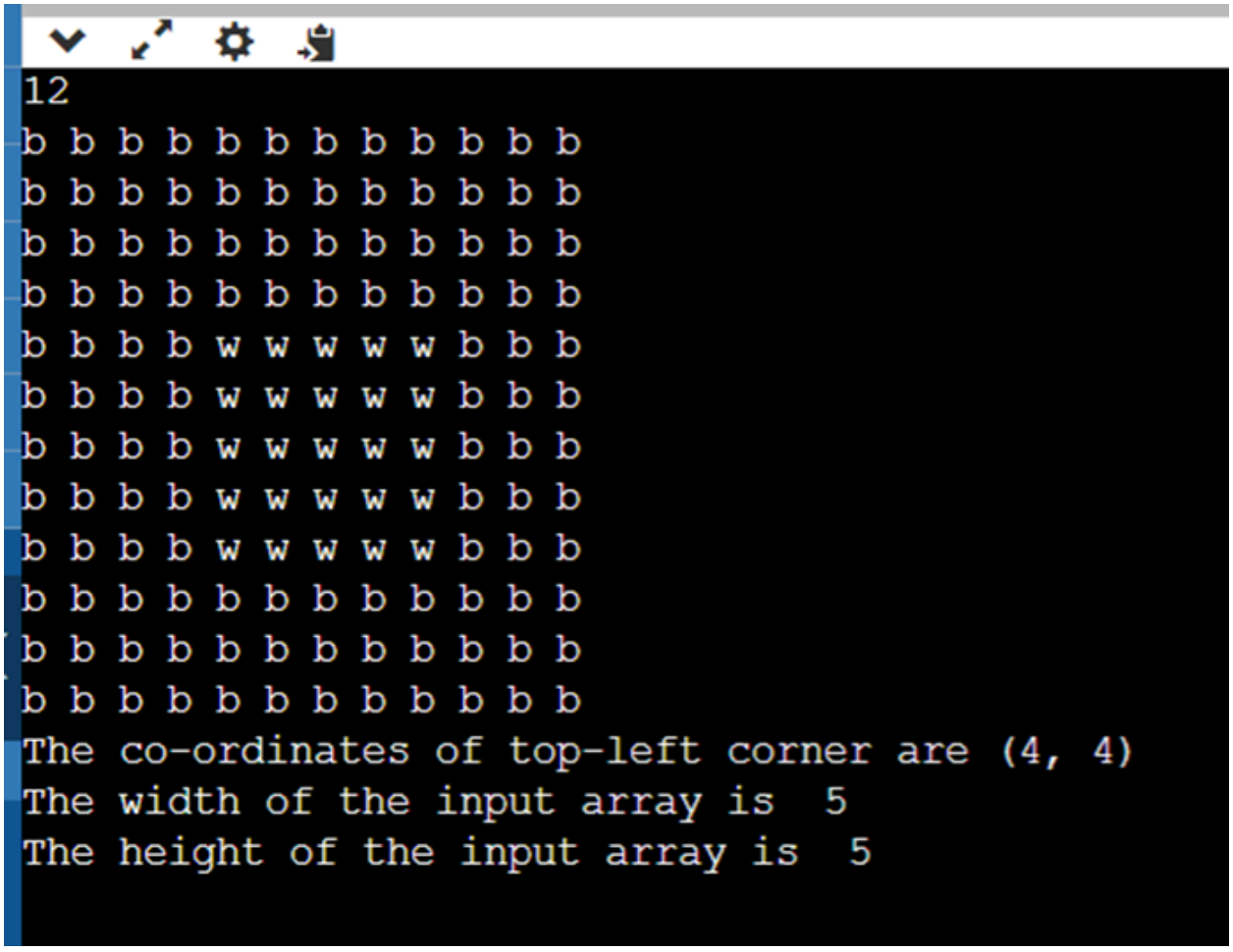
Code 1:

```
a=[]
n= int(input())
for i in range(n):
        l=list(map(str , input().split()))
        a.append(l)
k=[]
res = 0
for i in range(n):
        for j in range(n):
        if(a[i][j] == 'w'):
        k.append((i,j))
        res = 1
        break
        if (res == 1):
        break

# print(k)
if len(k) != 0:

        n1=k[0][1]
        c=0
        for i in range(n):
        if a[n1][i] == 'w':
```

```
        c+=1
        k.append(c) #width


        c1=0
        m=k[0][0]
        for j in range(n):



        if a[j][m] == 'w':
        c1+=1
        k.append(c1)# height
        print("The co-ordinates of top-left corner are", k[0])
        print("The width of the input array is ",k[1])
        print("The height of the input array is ",k[2])
        # print(k)
else:
        print("the list is empty")
```

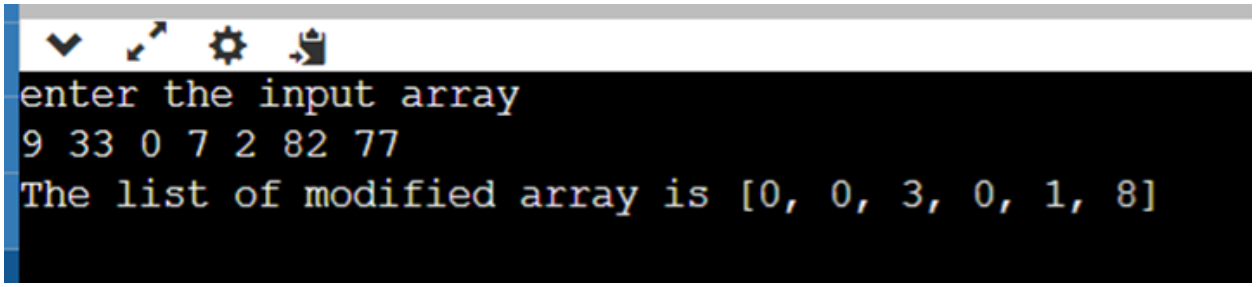
Output:




Code 2:

```
print("enter the input array")
l= list(map(int,input().split()))
k=[]
for i in range(len(l)-1):
        if l[i+1] == 0:
        continue
   k.append(l[i]//l[i+1])
```

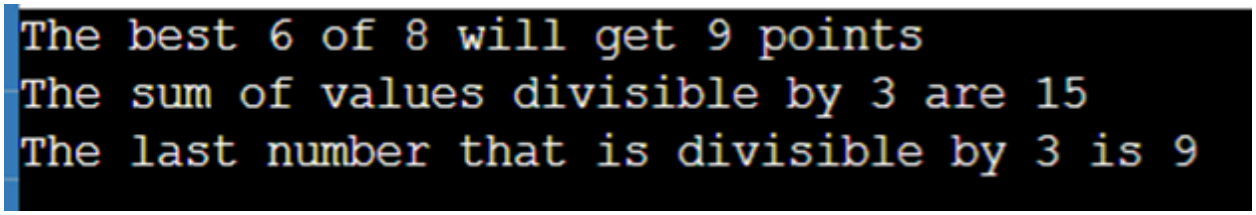
```
if len(k)!=0:


    if l[0] == 0:
print("The list of modified array is", k)
    else:
k.append(l[-1]//l[0])
print("The list of modified array is", k)
```




Code 3:

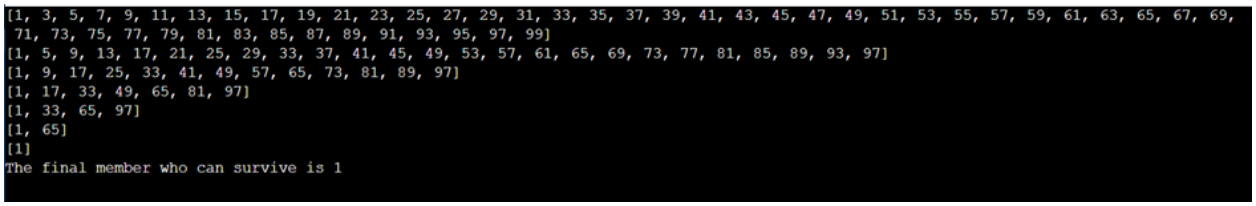
```
s = input()
k=0
last = -1
for i in s:
        if i.isdigit():
        res = int(i)
        if res%3 == 0:
        k+=res
        last = res
print("The sum of values divisible by 3 are", k)
```

```
if last == -1:
        print("there is no number that divisible by 3")
else:
        print("The last number that is divisible by 3 is",last)
```




Code 4:

```
l=[0]*100
for i in range(100):
        l[i]=i+1

while(len(l)>1):

        l=l[0::2]

        print(l)

print("The final member who can survive is" ,l[0])
```




Code 5:

```
-- Hotels Table
CREATE TABLE Hotels (
    hotel_id INT PRIMARY KEY,
    hotel_name VARCHAR(255),
    other_hotel_details VARCHAR(255)
);

-- Menus Table
CREATE TABLE Menus (
    menu_id INT PRIMARY KEY,
    hotel_id INT,
    menu_name VARCHAR(255),
    other_menu_details VARCHAR(255),
    FOREIGN KEY (hotel_id) REFERENCES Hotels(hotel_id)
);

-- Food Items Table
CREATE TABLE FoodItems (
    food_item_id INT PRIMARY KEY,
    food_item_name VARCHAR(255),
    other_food_item_details VARCHAR(255)
);

-- Menu-Food Items Relationship Table
CREATE TABLE MenuFoodItems (
    menu_id INT,
    food_item_id INT,
    FOREIGN KEY (menu_id) REFERENCES Menus(menu_id),
    FOREIGN KEY (food_item_id) REFERENCES FoodItems(food_item_id),
    PRIMARY KEY (menu_id, food_item_id)
);
```