

## Week-05-Nested Loops - while and for, Jumps in Loops

### Week-05-01-Practice Session-Coding

Question 1

Correct

Marked out of  
3.00

 Flag question

Write a program that prints a simple chessboard.

Input format:

The first line contains the number of inputs T.

The lines after that contain different values for size of the chessboard

Output format:

Print a chessboard of dimensions size \* size. Print a W for white spaces and B for black spaces.

Input:

2  
3  
5

Output:

WBW  
BWB  
WBW  
WBWBW  
BWBWB  
WBWBW  
BWBWB  
WBWBW

---

[Source code](#)

```
1 #include<stdio.h>
2 int main()
3 {
4     int t,arr[100];
5     scanf("%d",&t);
6     for(int i=0;i<t;i++)
7     {
8         scanf("%d",&arr[i]);
9     }
10    for(int z=0;z<t;z++)
11    {
12        for(int j=0;j<arr[z];j++)
13        {
14            for(int i=0;i<arr[z];i++)
15            {
16                if((i+j)%2==0)
17                {printf("W");}
18                else
19                {printf("B");}
20            }
21            printf("\n");
22        }
23    }
24    return 0;
25 }
```

---

Result

	<b>Input</b>	<b>Expected</b>	<b>Got</b>	
✓	2	WBW	WBW	✓
	3	BWB	BWB	
	5	WBW	WBW	
		WBWBW	WBWBW	
		BWBWB	BWBWB	
		WBWBW	WBWBW	
		BWBWB	BWBWB	
		WBWBW	WBWBW	

Passed all tests! ✓

### Question 2

Correct

Marked out of  
5.00

Flag question

Let's print a chessboard!

Write a program that takes input:

The first line contains T, the number of test cases

Each test case contains an integer N and also the starting character of the chessboard

Output Format

Print the chessboard as per the given examples

Sample Input / Output

Input:

```
2
2 W
3 B
```

Output:

```
WB
BW
BWB
WBW
BWB
```

[Source code](#)

```

1 #include<stdio.h>
2 int main()
3 {
4     int t,arr[100];
5     char ch[100];
6     scanf("%d",&t);
7     for(int i=0;i<t;i++)
8     {
9         scanf("%d %c",&arr[i],&ch[i]);
10    }
11    for(int z=0;z<t;z++)
12    {
13        for(int j=0;j<arr[z];j++)
14        {
15            for(int i=0;i<arr[z];i++)
16            {
17                if((i+j)%2==0)
18                {printf("%c",ch[z]);}
19                else{
20                    if(z>0)
21                    {printf("%c",ch[z-1]);}
22                    else
23                    {printf("%c",ch[z+1]);}
24                }
25            }
26            printf("\n");
27        }
28    }
29    return 0;
30 }
31

```

### Result

	<b>Input</b>	<b>Expected</b>	<b>Got</b>	
✓	2	WB	WB	✓
	2 W	BW	BW	
	3 B	BWB	BWB	
		WBW	WBW	
		BWB	BWB	

Passed all tests! ✓

**Question 3**

Correct

Marked out of  
7.00[Flag question](#)

Decode the logic and print the Pattern that corresponds to given input.

If N= 3

then pattern will be :

10203010011012

\*\*4050809

\*\*\*\*607

If N= 4, then pattern will be:

1020304017018019020

\*\*50607014015016

\*\*\*809012013

\*\*\*\*\*10011

Constraints

2 <= N <= 100

Input Format

First line contains T, the number of test cases

Each test case contains a single integer N

Output

First line print Case #i where i is the test case number

In the subsequent line, print the pattern

---

[Source code](#)

```
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     for(int i=1;i<=n;i++)
7     {
8         int a;
9         scanf("%d",&a);
10        int l=1,s=a,t=(a*(a+1))-a+1;
11        printf("Case #%d\n",i);
12        for(int j=0;j<a;j++)
13        {
14            int k=2*j,t1=t;
15            while(k>0)
16            {
17                printf("%c",'*');
18                k-=1;
19            }
20            for(int p=0;p<s;p++)
21            {
22                printf("%d",l);
23                l+=1;
24                printf("%d",0);
25            }
26            for(int q=0;q<s;q++)
27            {
28                printf("%d",t1);
29                t1+=1;
30                if(q==(s-1))
31                {
32                    break;
33                }
34                printf("%d",0);
35            }
36            s-=1;
37            t-=s;
38            printf("\n");
39        }

```

## Result

	<b>Input</b>	<b>Expected</b>	<b>Got</b>	
✓	3 3 4 5	Case #1 10203010011012 **4050809 ****607 Case #2 1020304017018019020 **50607014015016 ****809012013 *****10011 Case #3 102030405026027028029030 **6070809022023024025 ****10011012019020021 *****13014017018 *****15016	Case #1 10203010011012 **4050809 ****607 Case #2 1020304017018019020 **50607014015016 ****809012013 *****10011 Case #3 102030405026027028029030 **6070809022023024025 ****10011012019020021 *****13014017018 *****15016	✓

Passed all tests! ✓