WEEK:05

NESTED LOOPS – WHILE AND FOR, JUMPS IN LOOPS

WEEK: 05-01

ROLL NO: 240801161

NAME: KEERTHANA S

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Thursday, 19 December 2024, 9:31 AM
Duration	4 days 8 hours

QUESTION:1

SIMPLE CHESSBOARD

Problem Statement:

Write a program that prints a simple chessboard.

Input format:

The first line contains the number of inputs T.

The lines after that contain a different value for size of the chessboard

Output format:

Print a chessboard of dimensions size * size.

Print W for white spaces and B for black spaces.

Sample Input:

2

3

5

Sample Output:

WBW

BWB

WBW

WBWBW

BWBWB

WBWBW

BWBWB

WBWBW

PROGRAM:

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
int main ()
{
    int t,d,i=0,i1,i2,x;
    char c;
    scanf ("%d",&t);
    while (i<t)
    {
        scanf ("%d",&d);
        i1=0;
        while (i1<d)
        {

            x=1;
            i2=0;
            if (i1%2==0)
            {
                x=0;
            }
            while (i2<d)
            {
                c = 'B';
                if (i2%2==x)
                {
                    c='W';
                }
                printf("%c",c);
                i2++;

            }
            i1+=1;
           printf ("\n");

        }
        i=i+1;
    }
        return 0;

}
```

OUTPUT:

	Input	Expected	Got	
✓	2 3 5	WBW BWB WBW WBWBW BWBWB WBWBW BWBWB WBWBW	WBW BWB WBW WBWBW BWBWB WBWBW BWBWB WBWBW	✓

Passed all tests! ✓

QUESTION:2

PRINT OUR OWN CHESSBOARD

Problem Statement:

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:

2

2 W

3 B

Sample Output:

WB

BW

BWB

WBW

BWB

PROGRAM:

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main ()
{
    int t, d , i , i1, i2, o, z;
    char c,s;
    scanf("%d",&t);
    for (i=0;i<t;i++)
    {
        scanf("%d %c",&d,&s);
        for(i1=0;i1<d;i1++)
        {
            z=(s=='W')?0:1;
            o=(i1%2==z)?0:1;
            for(i2=0;i2<d;i2++)
            {
                c= (i2%2 ==o)?'W':'B';
                printf("%c",c);
            }
                printf("\n");


        }
    }
        return 0;
}
```

OUTPUT:

	Input	Expected	Got	
✓	2 2 W 3 B	WB BW BWB WBW BWB	WB BW BWB WBW BWB	✓

Passed all tests! ✓

QUESTION:3

PATTERN PRINTING

Problem Statement:

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 Format

First line print Case #i where i is the test case number, In the subsequent line, print the pattern

Sample Input

3

3

4

5

Sample Output

Case #1

10203010011012

**4050809

****607

Case #2

10203040170180190 20

**50607014015016

****809012013

******10011

Case #3

1020304050260270280290 30

**6070809022023024025

****10011012019020021

******13014017018

********15016

PROGRAM:

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int n,v,p3,c,in,i,i1,i2,t,ti;
    scanf("%d",&t);
    for(ti=0;ti<t;ti++)
    {
        v = 0;
        scanf("%d",&n);
        printf("Case #%d\n",ti+1);
        for (i=0;i<n;i++){
            c=0;
            if(i>0){
                for(i1=0;i1<i;i1++)
                printf("**");
            }
        for(i1=i;i1<n;i1++){
            if (i>0)
            c++;
            printf("%d0",++v);
        }
        if (i ==0)
        {
            p3=v+(v*(v-1))+1;
            in = p3;
        }
        in = in -c;
        p3= in;
        for(i2=i;i2<n;i2++){
        printf("%d",p3++);
        if(i2!=n-1)
        printf("0");
        }
        printf("\n");
      }
    }
    return 0;
}
```

OUTPUT:

	Input	Expected	Got	
✓	3	Case #1	Case #1	✓
	3	10203010011012	10203010011012	
	4	**4050809	**4050809	
	5	****607	****607	
		Case #2	Case #2	
		1020304017018019020	1020304017018019020	
		**50607014015016	**50607014015016	
		****809012013	****809012013	
		******10011	******10011	
		Case #3	Case #3	
		10203040502602702802 9030	10203040502602702802 9030	
		**6070809022023024025	**6070809022023024025	
		****10011012019020021	****10011012019020021	
		******13014017018	******13014017018	
		********15016	********15016	

Passed all tests! ✓