Question 1

Question text

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 values for size of the chessboard

Output format:

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

input:

2
3
5

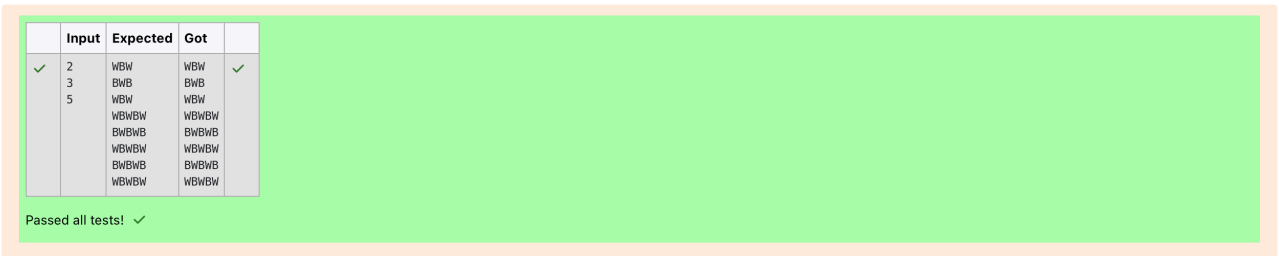
Output:

WBW
BWB
WBW
WBWBW
BWBWB
WBWBW
BWBWB
WBWBW

Program:

```
#include <stdio.h>
int main() {
        int n,t;
        scanf("%d",&n);
        while (n) {
                scanf("%d",&t);
                for (int i = 1; i <=t; i++) {
                        for (int j = 1; j<=t; j++) {
                                if ((i+j)%2 == 0)
                                printf("W");
                                else
                                printf("B");
                        }
                        printf("\n");
                }
                n—;
        }
        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

Question text

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

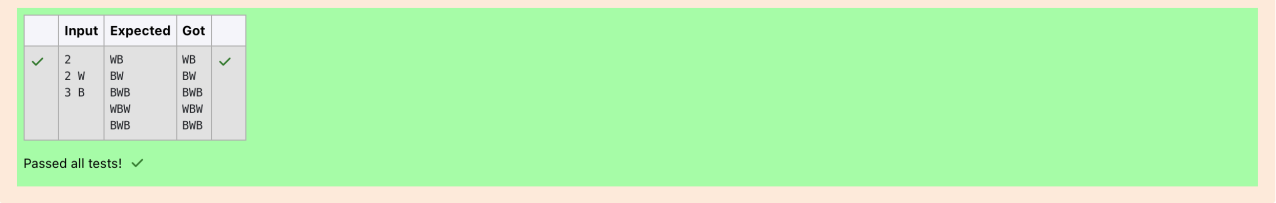
Output Format
Print the chessboard as per the given examples
Sample input / Output

input:
2
2 W
3 B
Output:
WB
BW
BWB
WBW
BWB

Program:

```
#include <stdio.h>
int main() {
        int n,t;
        char ch;
        scanf("%d",&n);
        while(n) {
                scanf("%d %c",&t,&ch);
                for (int i = 1; i <=t; i++) {
                        for (int j = 1; j<=t; j++) {
                                if ((j+i)%2 == 0)
                                printf("%c",ch);
                                else
                                printf("%ch",ch == 'W' ? 'B':'W');
                        }
                        printf("\n");
                }
                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

Question text
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

Test Case 1
3
3
4
5

Output
Case #1
10203010011012
**4050809
****607
Case #2
1020304017018019020
**50607014015016
****809012013
******10011
Case #3
102030405026027028029030
**6070809022023024025
****10011012019020021
******13014017018
********15016

Program:

```
#include <stdio.h>
#include <string.h>
int sum(int n) {
	return n*(n-1)/2;
}

void BSpattern(int N)
{
	int val = 0, Pthree = 0, c = 0, initial;
	char s[100] = "**";
	for (int i = 0; i < N; i++) {
		c = 0;
		if (i > 0) {
			printf("%s", s);
			strcat(s, "**");
		}
		for (int j = i; j < N; j++) {
			if (i > 0) {
				c++;
			}
			printf("%d", ++val);
			printf("0");
		}
		if (i == 0) {
			int sumb = sum(val) *
			Pthree = val + sumb + 1;
```

```
                initial = Pthree;
            }
            initial = initial - c;
            Pthree = initial;
            for (int k = i; k < N; k++) {
                printf("%d", Pthree++);
                if (k != N - 1) {
                    printf("0");
                }
            }
            printf("\n");
        }
}

int main()
{
        int N;
        scanf(“%d”,&N);
        for (int i = 1;i <= N; i++) {
            int Num;
            scanf(“%d”,&Num);
            printf(“Case#%d\n”,i);
            BSpattern{Num);
        }
        return 0;
}
```

Output:

	Input	Expected	Got	
✓	3 3 4 5	Case #1 10203010011012 **4050809 ****607 Case #2 10203040170180190 20 **50607014015016 ****809012013 ******10011 Case #3 10203040502602702802903 0 **607080902202302402 5 ****100110120190200 21 ******13014017018 ********15016	Case #1 10203010011012 **4050809 ****607 Case #2 10203040170180190 20 **50607014015016 ****809012013 ******10011 Case #3 10203040502602702802903 0 **607080902202302402 5 ****100110120190200 21 ******13014017018 ********15016	✓

Passed all tests! ✓