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

```c
1   #include<stdio.h>
2   int main(){
3       int T,d,i=0,i1,i2;
4       char c;
5       scanf("%d",&T);
6       while(i<T)
7       {
8           scanf("%d",&d);
9           i1=0;
10          while(i1<d)
11          {
12              i2=0;
13              while(i2<d)
14              {
15                  c='B';
16                  if((i1+i2)%2==0)
17                  {
18                      c='W';
19                  }
20                  printf("%c",c);
21                  i2++;
22              }
23              i1+=1;
24              printf("\n");
25
26          }
27          i=i+1;
28      }
29  }
```
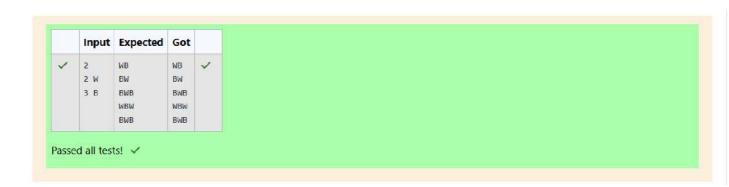
	Input	Expected	Got	
✓	2	WBW	WBW	✓
	3	BWB	BWB	
	5	WBW	WBW	
		WBWBW	WBWBW	
		BWBWB	BWBWB	
		WBWBW	WBWBW	
		BWBWB	BWBWB	
		WBWBW	WBWBW	

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

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int v;
    scanf("%d",&v);
    while(v!=0)
    {
        char a;
        int x;
        scanf("%d %c",&x,&a);
        for (int i=0;i<x;i++)
        {
            for(int j=0;j<x;j++)
            {
                printf("%c",a);
                if (a=='W')
                {
                    a='B';
                }
                else
                {
                    a='W';
                }
            }
            if((x%2)==0)
            {
                if(a=='W')
                a='B';
                else
                a='W';
            }
            printf("\n");


        }
        v--;
    }
}
```

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

Passed all tests! ✓

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

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

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main(){
    int v,c=0;
    scanf("%d",&v);
    while(v!=0){
        c++;
        int a;
        scanf("%d",&a);
        int s1=10,s2=(a*a*10)+10;
        printf("Case #%d\n",c);
        for(int i=0;i<a;i++){
            for(int j=0;j<i;j++){
                printf("**");
            }
            for(int j=0;j<a-i;j++){
                printf("%d",s1);
                s1+=10;
            }
            for(int j=0;j<a-i;j++){
                if((j+1)==(a-i)){
                    printf("%d",((s2+(j*10))/10));
                }
                else{
                    printf("%d",(s2+(j*10)));
                }
            }
            s2-=(a-i)*10;
            s2+=10;
            printf("\n");
        }
        v--;
    }
}
```

	Input	Expected	Got	
✓	3 3 4 5	Case #1 10203010011012 **4050809 ****607 Case #2 1020304017018019020 **50607014015016 ****809012013 ******10011 Case #3 10203040405026027028029030 **6070809022023024025 ****10011012019020021 ******13014017018 ********15016	Case #1 10203010011012 **4050809 ****607 Case #2 1020304017018019020 **50607014015016 ****809012013 ******10011 Case #3 10203040405026027028029030 **6070809022023024025 ****10011012019020021 ******13014017018 ********15016	✓

Passed all tests! ✓