Status	Finished
Started	Monday, 3 November 2025, 8:16 PM
Completed	Monday, 3 November 2025, 8:50 PM
Duration	34 mins 18 secs

Question **1**

Correct

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

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
int main()
{
    int T,size;
    scanf("%d",&T);
    while(T--)
    {
        scanf("%d",&size);
        for(int i=0;i<size;i++)
{
    for(int j=0;j<size;j++)
    {
                if((i+j)%2==0)
                    printf("W");
                else
```

```
                    printf("B");
    }
    printf("\n");
}
    }
    return 0;
}
```

	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**

Correct

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 t,n;
    char c;
    scanf("%d",&t);
    while(t--)
    {
        scanf("%d %c",&n,&c);
        for(int i=0;i<n;i++)
        {
            for(int j=0;j<n;j++)
            {
                if((i+j)%2==0)
                    printf("%c",c);
                else
```

```
17                      printf("%c",c== 'W' ? 'B' : 'W');
18              }
19              printf("\n");
20          }
21      }
22      return 0;
23  }
```

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

Passed all tests! ✓

Question **3**

Correct

Problem Statement:

In a small coding competition, participants are to be grouped into teams of three members, each member represented by a number — 1, 2, and 3.
The rule of the competition states that no member can repeat within the same team.

Write a program to display all possible unique team combinations that can be formed using the members 1, 2, and 3 without repetition.

Sample Output:

1 2 3

1 3 2

2 1 3

2 3 1

3 1 2

3 2 1

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
int main()
{
    int i,j,k;
    for(i=1;i<=3;i++)
    {
        for(j=1;j<=3;j++)
        {
            for(k=1;k<=3;k++)
            {
                if(i!=j&&j!=k&&i!=k)
                printf("%d %d %d\n",i,j,k);
            }
        }
    }
    return 0;
}
```

	Expected	Got	
✓	1 2 3 1 3 2 2 1 3 2 3 1 3 1 2 3 2 1	1 2 3 1 3 2 2 1 3 2 3 1 3 1 2 3 2 1	✓

Passed all tests! ✓