Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Monday, 25 November 2024, 10:59 PM
Duration	27 days 18 hours
Correct	Write a program that prints a simple chessboard.
Marked out of 3.00  Flag question	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:

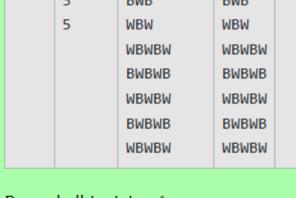
WBW
BWB
WBWBW
BWBWB
WBWBW

**BWBWB** 

**WBWBW** 

```
1 #include<stdio.h>
 2 v int main(){
         int t;
 3
        scanf("%d",&t);
 4
        for (int i=0;i<t;i++){</pre>
 5 v
             int size;
 6
             scanf("%d",&size);
             for(int j=0;j<size;j++){</pre>
 8 🔻
                 for(int k=0;k<size;k++){</pre>
 9 🔻
                      if((j+k)\%2==0){
10 v
                          printf("W");
11
                      }else{
12 v
13
                          printf("B");
14
15
                 printf("\n");
16
17
18
19
         return 0;
20
```

	Input	Expected	Got	
~	2	WBW	WBW	,
	3	BWB	BWB	
	5	WBW	WBW	
		WBWBW	WBWBW	
		BWBWB	BWBWB	
		WBWBW	WBWBW	
		BWBWB	BWBWB	
		WBWBW	WBWBW	



Passed all tests! 

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

Question 2

Marked out of

Flag question

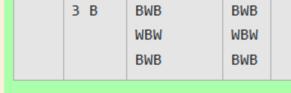
Correct

5.00

BWB WBW BWB

```
#include<stdio.h>
    int main(){
 3
        int T;
        int n;
        char m;
        scanf(" %d",&T);
 6
 7 ,
        for(int i=0;i<T;i++){</pre>
             scanf(" %d",&n);
 8
             scanf(" %c",&m);
             for(int j=0;j<n;j++){</pre>
10
                 for(int k=0;k<n;k++){</pre>
11 v
                     if((j+k)\%2==0){
12 1
13
                          printf("%c",m);
                      }else{
14 •
15 v
                          if(m=='W'){
                              printf("B");
16
                          }else{
17
                              printf("W");
18
19
20
21
                 printf("\n");
22
23
24
25
        return 0;
26
```

	Input	Expected	Got	
~	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

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
```

```
#include<stdio.h>
 2 v int main(){
         int n,v,p3,c,in,i,i1,i2,t,ti;
 3
        scanf("%d",&t);
 4
        for(ti=0;ti<t;ti++){</pre>
 5 v
 6
             v=0;
             scanf("%d",&n);
             printf("Case #%d\n",ti+1);
 8
             for(i=0;i<n;i++){</pre>
 9 ,
10
                c=0;
                 if(i>0){
11 v
                  for (i1=0;i1<i;i1++) printf("**");</pre>
12
13
             for(i1=i;i1<n;i1++){</pre>
14 ▼
                if(i>0) c++;
15
                printf("%d0",++v);
16
17
```

```
if (i==0){
18 •
               p3=v+(v*(v-1))+1;
19
               in=p3;
20
21
            in-=c;
22
            p3=in;
23
            for(i2=i;i2<n;i2++){</pre>
24 ▼
                printf("%d",p3++);
25
                if(i2!=n-1) printf("0");
26
             }printf("\n");
27
28
29
30
31
```

	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	
		102030405026027028029030	102030405026027028029030	
		**6070809022023024025	**6070809022023024025	
		****10011012019020021	****10011012019020021	
		*****13014017018	*****13014017018	
		*******15016	*******15016	

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Friday, 22 November 2024, 5:33 AM
Duration	31 days 11 hours
Question 1  Correct  Marked out of 3.00  Flag question	The k-digit number N is an Armstrong number if and only if the k-th power of each digit sums to N.  Given a positive integer N, return true if and only if it is an Armstrong number.
	Example 1:

Input:

153

Output:

true

Explanation:

153 is a 3-digit number, and  $153 = 1^3 + 5^3 + 3^3$ .

Example 2:
Input:
123
Output:
false
Explanation:
123 is a 3-digit number, and 123 != 1^3 + 2^3 + 3^3 = 36.
Example 3:
Input:
1634
Output:
true

Note:

```
1 <= N <= 10^8
```

```
1 #include<stdio.h>
    #include<math.h>
 3 v int main(){
        int n,d,nt,AN,i;
 4
        scanf("%d",&n);
        nt=n;
 6
        for(d=0;nt!=0;d++){
 7 🔻
            nt=nt/10;
 8
 9
10
        nt=n;
        for(i=0;i<d;i++){</pre>
11 🔻
            AN=AN+pow(nt%10,d);
12
13
            nt=nt/10;
        }if(AN==n){
14 v
            printf("true");
15
16 •
        }else{
            printf("false");
17
18
        return 0;
19
20
21
```

	Input	Expected	Got	
~	153	true	true	
~	123	false	false	

Passed all tests! 

Question 2

Correct

Marked out of 5.00

Flag question

```
#include<stdio.h>
  int main(){
        int num,rev,par;
        scanf("%d",&num);
 4
        while(1){
            par=num;
 6
            rev=0;
            while(num>0){
8
                rev=rev*10+num%10;
 9
10
                num/=10;
11
12
            num=par+rev;
            int temp=num, l=0;
13
            while(temp>0){
14
                int digit=temp%10;
15
                l=l*10+digit;
16
                temp/=10;
17
            }if (num==1){
18
                break;
19
20
        }printf("%d",num);
21
22
```

	Input	Expected	Got
~	32	55	55
~	789	66066	66066

Passed all tests! 

Question **3** A number is considered lucky if it contains either 3 or 4 or 3 and 4 both in it. Write a program to print the nth lucky number. Example, 1st Correct lucky number is 3, and 2nd lucky number is 4 and 3rd lucky number is 33 and 4th lucky number is 34 and so on. Note that 13, 40 etc., are not Marked out of lucky as they have other numbers in it. 7.00 Flag question The program should accept a number 'n' as input and display the nth lucky number as output. Sample Input 1: 3 Sample Output 1: 33 Explanation: Here the lucky numbers are 3, 4, 33, 34., and the 3rd lucky number is 33. Sample Input 2: 34 Sample Output 2:

```
1 #include<stdio.h>
 2 v int main(){
 3
        int n;
        scanf("%d",&n);
 4
        int count=0;
 5
        int num=3;
 6
        while(count<n){</pre>
7 🔻
            int a=0;
 8
            int temp=num;
 9
            int b=0;
10
            while (temp>0){
11 ▼
                 int digit=temp%10;
12
                 if(digit==3||digit==4){
13 v
14
                     a=1;
15 v
                 }else{
16
                     b=1;
                     break;
17
18
19
                temp/=10;
20 •
            }if(a&&!b){
21
                 count ++;
22 🔻
                 if (count==n){
                     printf("%d",num);
23
24
25
             }num++;
26
```

Input Expected Got 33344 🗸 33344 Passed all tests!