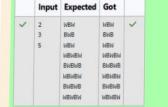
Question 1	Write a program that prints a simple chessboard.
Marked out of 3.00	Input format:
Flag question	input tornies.
	The first line contains the number of inputs T.
	The lines after that contain a different values for size of the chessboard
	Output format:
	Print a chessboard of dimensions size * size. Print a Print W for white spaces and B for black spaces.
	Input
	2
	3
	Output:
	WBW
	BWB
	WBW
	WBWBW
	BWBWB WBW
	BWBWB
	WBWBW

August (populty regimes 0.96)





Passed all tests! V

Question 2 Let's print a chessboard! Correct Marked out of 5.00 Write a program that takes input: F Flag question 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 W 3 B Output: WB BW BWB WBW BWB Answer: (penalty regime: 0 %)



	Input	Expected	Got	
/	2	WB	WB	~
	2 W	BW	BW	
	3 B	BWB	BWB	
		WBW	WBW	
		BWB	BWB	

Passed all tests! <

.

Question 3 Correct	Decode the logic and print the Pattern that corresponds to given input.
Marked out of 7.00	If N= 3
Flag question	
	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

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

Answer: (penalty regime: 0 %)



	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	

Passed all tests! <

Question 1	The k-digit number N is an Armstrong number if and only if the k-th power of each digit sums to N.	
Correct Marked out of 3.00	Given a positive integer N, return true if and only if it is an Armstrong number.	
₹ Flag question	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:	

```
1634
Output:
true
Note:
1 <= N <= 10^8
Answer: (penalty regime: 0 %)
   1 #include(stdio.h)
   2 #include<math.h>
   3 + int main(){
   4
          int n;
          scanf("%d",&n);
   5
          int x=0, n2=n;
   6
          while(n2!=0){
   8
              X++;
              n2=n2/10;
   9
  10
  11
  12
          int sum=0,n4, n3=n;
          while(n3!=0){
  13
             n4=n3%10;
  14
             sum=sum+pow(n4,x);
  15
  16
             n3=n3/10;
  17
          (n==sum) ? printf("true"): printf("false");
  18
  19
 20
      Input Expected Got
    153
            true
                     true
                     false 🗸
     123
            false
Passed all tests! <
```

Input:



	Input	Expected	Got		
~	32	55	55	~	
~	789	66066	66066	~	

Passed all tests! ✓

Question 2

Marked out of 5.00

P Flag question

Correct

Question 3 Correct Marked out of 7.00	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 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 lucky as they have other numbers in it.
₹ 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:
	33344
	Answer: (penalty regime: 0 %)



Passed all tests! <