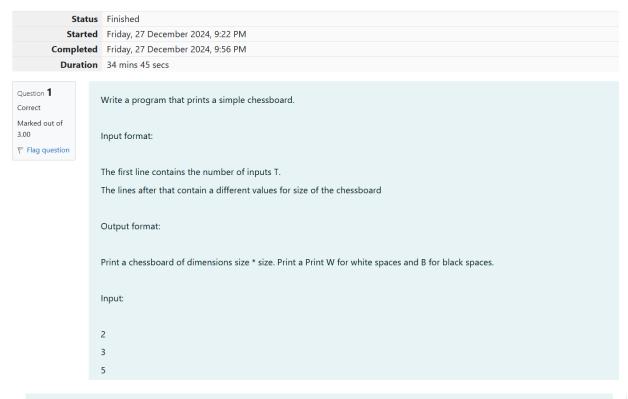
Week-05-Nested Loops - while and for, Jumps in Loops

Week-05-01-Practive Session-coding

ROLL NO: 241801294

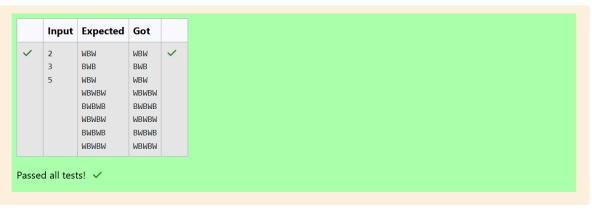
NAME: THARUN N

Q1)



Output:
WBW
BWB
WBW
WBWBW
BWBWB
WBWBW
BWBWB
WBWBW

```
Answer: (penalty regime. 0 %)
    1 #include<stdio.h>
    3 ,
       int main() {
   4
           int v;
   5
           scanf("%d", &v);
   6
           while (v > 0) {
   7
                int x;
   8
                scanf("%d", &x);
   9
                if (x < 0) {
   10
                   x = -x;
   11
  12
                char a ='W';
  13
                for (int i = 0; i < x; i++) {
                    for (int j = 0; j < x; j++) {
    printf("%c", a);</pre>
  14
  15
  16
                        if (a == 'W')
                            `a = 'B';
  17
   18
                        else
                            a ='W';
  19
  20
                    }
                    printf("\n");
  21
  22
  23
                    if (x%2 == 0) {
                        if (a == 'W')
   24
                            `a ='B';
  25
   26
                        else
  27
                            a = 'W';
  28
   29
   30
   31
           }
   32 }
```



Q2)

```
Question \mathbf{2}
                     Let's print a chessboard!
Correct
Marked out of
5.00
                     Write a program that takes input:

▼ 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
                     2 W
                     3 B
```

```
Output:
WB
BW
BWB
WBW
BWB
Answer: (penalty regime: 0 %)
   1 #include<stdio.h>
   3 ,
       int main() {
          int v;
scanf("%d", &v);
   4
   5
   6
         8
   9
  10
  11 ,
  12 •
  13
  14
  15
  16
  17
  18
  19
                 if ((x%2) == 0) {
    if ( a == 'W')
    a = 'B';
else
  20
  21
  22
  23
                     a = 'W';
  24
```

```
23 | else | a = 'W'; | 25 | printf("\n"); | } | v--; | 30 | }
```

2	WB	WB	~
2 W	BW	BW	
3 B	BWB	BWB	
	WBW	WBW	
	BWB	BWB	

Question **3**Correct
Marked out of 7.00

Flag question

2 <= N <= 100

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

```
1020304017018019020
**50607014015016
****809012013
*****10011
Case #3
102030405026027028029030
**6070809022023024025
****10011012019020021
*****13014017018
******15016
Answer: (penalty regime: 0 %)
  1 |#include<stdio.h>
    2
    3 🔻
         int main() {
            int v, c = 0;
scanf("%d", &v);
    4
    5
             while(v != 0) {
    6
                  C++;
    8
                  int a;
                  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("**");
    }
}</pre>
    9
   10
   11
   12
   13
   14
   15
                        }
   16
                        17
   18
```

```
Answer: (penalty regime: 0 %)
   1 #include<stdio.h>
   3 * int main() {
           int v, c = 0;
scanf("%d", &v);
   4
   5
   6 •
           while(v != 0) {
               C++;
   8
               int a;
               scanf("%d", &a);
   9
               int s1 = 10, s2 = (a*a*10)+10;
  10
                printf("Case #%d\n", c);
  11
                for (int i = 0; i < a; i++) {
  12 ,
                    for (int j = 0; j < i; j++) {
    printf("**");</pre>
  13
  14
  15
  16
                    for (int j = 0; j < a-i; j++) {
    printf("%d", s1);</pre>
  17
  18
  19
                        s1 += 10;
  20
  21
                    for (int j = 0; j < a-i; j++) {
  22
  23
                       if((j+1) == (a-i)) {
  24
                           printf("%d", ((s2+(j*10))/10));
                        } else
  25
                       printf("%d", (s2+(j*10)));
  26
  27
  28
                    s2 -= ((a-i)*10);
  29
  30
                    s2+=10;
  31
   32
                    printf("\n");
  33
   34
               V--;
   35
   36 }
```

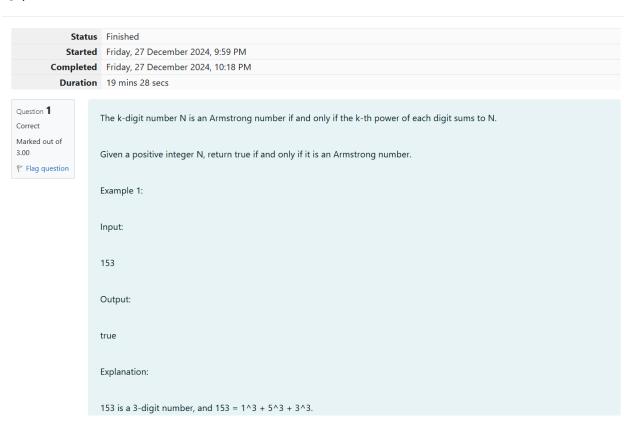
~	3	Case #1	Case #1	~
	3	10203010011012	10203010011012	
	4	**4050809	**4050809	
	5	****607	****607	
		Case #2	Case #2	
		1020304017018019020	1020304017018019020	
		**50607014015016	**50607014015016	
		****809012013	****809012013	
		*****10011	*****10011	
		Case #3	Case #3	
		102030405026027028029030	102030405026027028029030	
		**6070809022023024025	**6070809022023024025	
		****10011012019020021	****10011012019020021	
		*****13014017018	*****13014017018	
		*******15016	*******15016	

Week-05-02-Practive Session-coding

ROLL NO: 241801294

NAME: THARUN N

Q1)



```
Input:

123

Output:

false

Explanation:

123 is a 3-digit number, and 123 != 1^3 + 2^3 + 3^3 = 36.

Example 3:

Input:
```

```
Output:
true
Note:
1 <= N <= 10^8
Answer: (penalty regime: 0 %)
1 #include<stdio.h>
   3 v int main() {
          int a, b = 0, c, e = 0;
scanf("%d", &a);
   4
   5
   6
7
         if (a < 0)
         a = -a;
c = a;
   8
   9
          while (c != 0) {
  10 🔻
          c/=10;
e++;
  11
  12
  13
          }
          c = a;
  14
  15 🔻
          while (c != 0) {
          int d = c%10;
int f = 1;
for (int i = 0; i < e; i++) {
  16
  17
  18
  19
              f *= d;
  20
             b += f;
  21
```

```
int d = c%10;
int f = 1;
for (int i = 0; i < e; i++) {
   f *= d;
16
17
18 •
19
20
          b += f;
21
22
            c /= 10;
23
24
25 🔻
         if ( a == b) {
26
            printf("true");
27
         } else
28
            printf("false");
29 }
```

	Input	Expected	Got	
~	153	true	true	~
~	123	false	false	~

c /= 10;

Passed all tests! ✓

22

23

Question **2** Incorrect Marked out of 5.00

Flag question

Take a number, reverse it and add it to the original number until the obtained number is a palindrome. Constraints 1<=num<=99999999 Sample Input 1 32 Sample Output 1 55 Sample Input 2 789 Sample Output 2 66066

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
    int revNum(int a) {
        int b = 0, c = a;
while (c != 0) {
4
5
            b = (b*10)+(c%10);
            c/=10;
8
9
10
        return b;
11
12
13
    int palindrome(long a) {
14
        return (a == revNum(a));
15
16
17
   int main() {
18
19
        int a, b = 0, c;
20
        scanf("%d", &a);
21
22
        while (c != 0 ) {
23
            b = (b*10) + (c%10);
24
25
26
27
        while (palindrome(x) == 0) {
28
            x+=revNum(x);
29
30
        printf("%d", x);
31 }
```

Q3)

Question **3**Correct
Marked out of 7.00

Flag question

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.

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

Input Exped
34 33344
ed all tests! 🗸