

Week 5:

Write a C program to count total number of digits of an Integer number (N).

Sample Test Cases

Test Case 1

Input

3456

Output

The number 3456 contains 4 digits.

Test Case 2

Input

30000

Output

The number 30000 contains 5 digits.

Test Case 3

Input

57

Output

The number 57 contains 2 digits.

Test Case 4

Input

909

Output

The number 909 contains 3 digits.

For example:

Input Result

3456

The number 3456 contains 4 digits.

30000

The number 30000 contains 5 digits.

57

The number 57 contains 2 digits.

909

The number 909 contains 3 digits.

The image shows a C program in a code editor and its execution results. The program is designed to count the number of digits in a given integer. It uses a loop to divide the number by 10 until it reaches zero, incrementing a counter for each division. The program is tested with several inputs, and the results are shown in a table.

```
Answers: (penalty regime: 0 %)  
#include<stdio.h>  
int main()  
{  
    int n,digits=0,og;  
    scanf("%d",&n);  
    og=n;  
    for(n=n;n!=0;p=n/10)  
        digits++;  
    printf("The number %d contains %d digits.",og,digits);  
    return 0;  
}
```

Input	Expected	Got	
✓ 3456	The number 3456 contains 4 digits.	The number 3456 contains 4 digits.	✓
✓ 30000	The number 30000 contains 5 digits.	The number 30000 contains 5 digits.	✓
✓ 57	The number 57 contains 2 digits.	The number 57 contains 2 digits.	✓
✓ 909	The number 909 contains 3 digits.	The number 909 contains 3 digits.	✓

Your code failed one or more hidden tests.
Your code must pass all tests to earn any marks. Try again.

Write a C program to check whether the given number(N) can be expressed as Power of Two (2) or not.

For example, 8 can be expressed as 2^3 .

Sample Test Cases

Test Case 1

Input

8

Output

8 is a number that can be expressed as power of 2.

Test Case 2

Input

46

Output

46 cannot be expressed as power of 2.

REC-CIS

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main()
{
    int n;
    scanf("%d",&n);
    if((n&(n-1))==0)
        printf("%d is a number that can be expressed as power of 2.",n);
    else
        printf("%d cannot be expressed as power of 2.",n);
    return 0;
}

/* the logic here is all even nos consists of only one one in its binary form
for eg num 8=1000 and num-1 is 7=0111
using bitwise and eliminates the only one in the num making it zero.
this is unique for even hence its used as a condition*/
```

REC-CIS

/* the logic here is all even nos consists of only one one in its binary form
for eg num 8=1000 and num-1 is 7=0111
using bitwise and eliminates the only one in the num making it zero.
this is unique for even hence its used as a condition*/

Input	Expected	Got
✓ 8	8 is a number that can be expressed as power of 2.	8 is a number that can be ex
✓ 46	46 cannot be expressed as power of 2.	46 cannot be expressed as po
✓ 1024	1024 is a number that can be expressed as power of 2.	1024 is a number that can be

Passed all tests! ✓

Write a program in C to find the sum of the series $1 + 11 + 111 + 1111 + \dots + n$ terms (n will be given as input from the user and sum will be the output)

Sample Test Cases

Test Case 1

Input

4

Output

1234

Test Case 2

Input

6

Output

123456

The screenshot displays a web browser window with a coding interface. The browser's address bar shows a URL from 'rajalakshmi-college.org'. The page title is 'Coding: Answer review (ABC)'. Below the browser window, there is a code editor with the following C code:

```
Answer: (penalty regime: 0 %)  
#include<stdio.h>  
int main()  
{  
    int terms,i,sum=0,n=1;  
    scanf("%d",&terms);  
    for(i=1;i<=terms;i++)  
    {  
        sum=sum+n;  
        n=n*10+i;  
    }  
    printf("%d",sum);  
    return 0;  
}
```

Below the code editor, there is a table showing test results:

	Input	Expected	Got	
✓	4	1234	1234	✓
✓	6	123456	123456	✓

Below the table, it says "Passed all tests! ✓". At the bottom of the page, there is a button labeled "Save the state of the flags" and another button labeled "Finish review".