[**04 - Iteration Control Structures**](https://www.rajalakshmicolleges.net/moodle/course/view.php?id=84#section-4)

**Ex. No. : 4.1 Date: 12.04.24**

**Register No.: 231401110** **Name: Subash.G**



# Nth Fibonacci

Write a [program](https://www.rajalakshmicolleges.net/moodle/mod/quiz/view.php?id=3478) to return the nth number in the fibonacci series. The value of N will be passed to the [program](https://www.rajalakshmicolleges.net/moodle/mod/quiz/view.php?id=3478) as input.

NOTE: Fibonacci series looks like –

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, . . . and so on.

i.e. Fibonacci series starts with 0 and 1, and continues generating the next number as the sum of the previous two numbers.

* first Fibonacci number is 0,
* second Fibonacci number is 1,
* third Fibonacci number is 1,
* fourth Fibonacci number is 2,
* fifth Fibonacci number is 3,
* sixth Fibonacci number is 5,
* seventh Fibonacci number is 8, and so on.

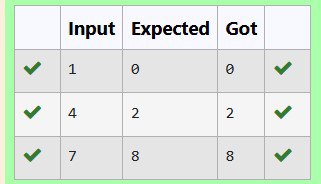
**For example:**

| **Input** | **Result** |
| --- | --- |
| 1  4  7 | 0  2  8 |

**Program:**

a=int(input())b=0 c=1 if(a==1): print("0") elif(a==2): print("1") else: for i in range (3,a+1):

d=b+c b=c c=d print(d)



**Ex. No. : 4.2 Date: 12.04.24**

**Register No.: 231401110** **Name: Subash.G**



[**Factors of a number**](https://www.rajalakshmicolleges.net/moodle/mod/quiz/view.php?id=5720)

Determine the factors of a number (i.e., all positive integer values that evenly divide into a number).

**For example:**

| **Inpu**  **t** | **Result** |
| --- | --- |
| 20 | 1 2 4 5 10  20 |

**Program:**

a=int(input()) for i in range(1,a+1): if(a%i==0):

print(i,end=" ")



**Ex. No. : 4.3 Date: 12.04.24**

**Register No.: 231401110** **Name: Subash.G**



# Product of single digit

Given a positive integer N, check whether it can be represented as a product of single digit numbers.

Input Format:

Single Integer input.

Output Format:

Output displays Yes if condition satisfies else prints No.

Example Input:

14

Output:

Yes

Example Input:

13

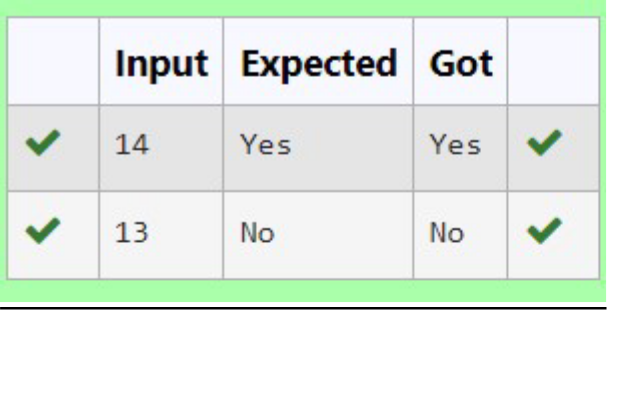
Output:

No **Program:**

a=int(input()) c=0 for i in range(1,10): for j in range(1,10):

if i\*j==a: c=1 if(c==1): print("Yes") ▾else:

print("No")



**Ex. No. : 4.4 Date: 12.04.24**

**Register No.: 231401110** **Name: Subash.G**



# Unique Digit Count

Write a program to find the count of unique digits in a given number N. The number will be passed to the program as an input of type int.

Assumption: The input number will be a positive integer number >= 1 and <= 25000.

For e.g.

If the given number is 292, the program should return 2 because there are only 2 unique digits '2' and '9' in this number

If the given number is 1015, the program should return 3 because there are 3 unique digits in this number, '1', '0', and '5'.

**For example:**

| **Input** | **Result** |
| --- | --- |
| 292 | 2 |
| 1015 | 3 |

**Program:**

a=input() b=len(set(a)) print(b)



**Ex. No. : 4.5 Date: 12.04.24**

**Register No.: 231401110** **Name: Subash.G**

# [Non](https://www.rajalakshmicolleges.net/moodle/mod/quiz/view.php?id=5717) Repeated Digit Count

Write a program to find the count of non-repeated digits in a given number N. The number will be passed to the program as an input of type int.

Assumption: The input number will be a positive integer number >= 1 and <= 25000.

Some examples are as below.

If the given number is 292, the program should return 1 because there is only 1 nonrepeated digit '9' in this number

If the given number is 1015, the program should return 2 because there are 2 nonrepeated digits in this number, '0', and '5'.

If the given number is 108, the program should return 3 because there are 3 non-repeated digits in this number, '1', '0', and '8'.

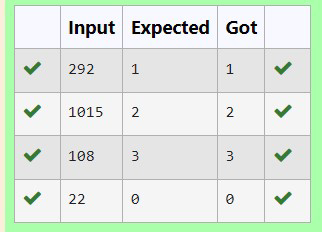
If the given number is 22, the function should return 0 because there are NO nonrepeated digits in this number.

**For example:**

| **Input** | **Resul**  **t** |
| --- | --- |
| 292 | 1 |
| 1015 | 2 |
| 108 | 3 |
| 22 | 0 |

**Program:** a={}

for i in input: if i in a:a[i]+=1 else:a[i]=1 print(sum([1 for i in a if a[i]==1]))



**Ex. No. : 4.6 Date: 12.04.24**

**Register No.: 231401110** **Name: Subash.G**

# Next Perfect Square

Given a number N, find the next perfect square greater than N.

Input Format:

Integer input from stdin.

Output Format:

Perfect square greater than N.

Example Input:

10

Output:

16

**Program:** import math a=int(input()) b = a + 1 while b > 0 :

m=math.sqrt(b) if(m==int(m)):

print(b) break else:



**Ex. No. : 4.7 Date: 12.04.24**

**Register No.: 231401110** **Name: Subash.G**

# Sum of Series

Write a program to find the sum of the series 1 +11 + 111 + 1111 + . . . + 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

Explanation:

as input is 4, have to take 4 terms.

1 + 11 + 111 + 1111

Test Case 2

Input

6

Output

123456

**For example:**

| **Input** | **Result** |
| --- | --- |
| 3 | 123 |

**Program:**

a=int(input()) t=1 s=0 for i in range(a) s+=t t=t\*10+1



**Ex. No. : 4.8 Date: 12.04.24**

**Register No.: 231401110** **Name: Subash.G**

# Prime Checking

Write a program that finds whether the given number N is Prime or not. If the number is prime, the program should return 2 else it must return 1.

Assumption: 2 <= N <=5000, where N is the given number.

Example1: if the given number N is 7, the method must return 2 Example2: if the given number N is 10, the method must return 1 **For example:**

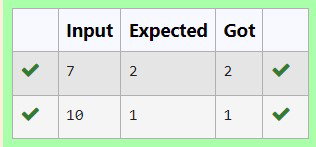
| **Input** | **Result** |
| --- | --- |
| 7 | 2 |
| 10 | 1 |

**Program:**

a=int(input()) c=0 for i in range(2,a):

if(a%i==0):

c=1 if(c==1): print("1") elif(c==0): print("2”)



**Ex. No. : 4.9 Date: 12.04.24**

**Register No.: 231401110** **Name: Subash.G**



# Disarium Number

A Number is said to be Disarium number when the sum of its digit raised to the power of their respective positions becomes equal to the number itself. Write a [program](https://www.rajalakshmicolleges.net/moodle/mod/quiz/view.php?id=3478) to print number is Disarium or not.

Input Format:

Single Integer Input from stdin.

Output Format:

Yes or No.

Example Input:

175

Output:

Yes

Explanation 1^1 + 7^2 +5^3 = 175 Example Input:

123

Output:

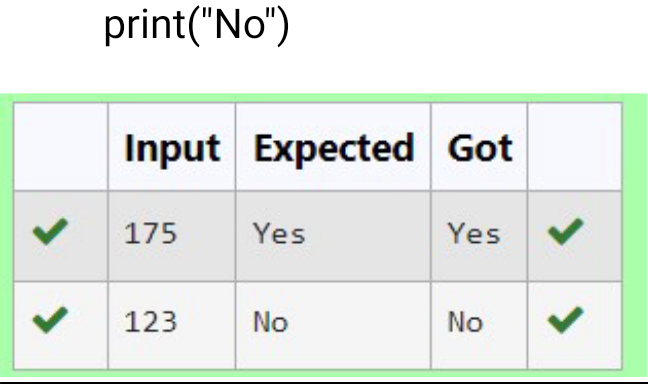
No

**For example:**

| **Inpu**  **t** | **Resul**  **t** |
| --- | --- |
| 175 | Yes |
| 123 | No |

**Program:**

a=input() n=len(a) r=0 for i,d in enumerate(a): r+=int(d)\*\*(i+1) if r==int(a): print("Yes") else:



**Ex. No. : 4.10 Date: 12.04.24**

**Register No.: 231401110** **Name: Subash.G**



**Perfect Square After adding One**

Given an integer N, check whether N the given number can be made a perfect square after adding 1 to it.

Input Format:

Single integer input.

Output Format:

Yes or No.

Example Input:

24

Output:

Yes

Example Input:

26

Output:

No

**For example:**

| **Input** | **Resul**  **t** |
| --- | --- |
| 24 | Yes |

**Program:**

import math a=int(input()) b=a+1 c=math.sqrt(b) if(c==int(c)): print("Yes") else:

