04 - Iteration Control Structures

Ex. No. : 4.1 **Date: 16/03/24**

Register No.: 230701348 **Name: N SUBRAMANIAN**

Factors of a number

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

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


For example:

Input	Result
20	1 2 4 5 10 20

Ex. No. : 4.2 **Date: 16/03/24**

Register No.: 230701348 **Name: N SUBRAMANIAN**

Non 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 non-repeated digit '9' in this number
If the given number is 1015, the program should return 2 because there are 2 non-repeated 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 non-repeated digits in this number.

```
a=int(input())
t=a
co=0
while t>0:
        t//=10
        co+=1
for i in str(a):
        c=0
        for j in str(a):
                if i==j:
                        c+=1
        if c>1:
                co-=1
print(co)
```

For example:

Input	Result
292	1
1015	2
108	3
22	0

Ex. No. : 4.3 **Date: 16/03/24**

Register No.: 230701348 **Name: N SUBRAMANIAN**

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.

```
a=int(input())
if a==1:
    print(1)
elif a==2:
    print(2)
else:
    flag=0
    for i in range(2,a):
        if a%i==0:
            flag=1
    if flag:
        print(1)
    else:
        print(2)
```


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

Ex. No. : 4.4 **Date: 16/03/24**

Register No.: 230701348 **Name: N SUBRAMANIAN**

Next Perfect Square

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

```
a=int(input())
while 1:
    for i in range(1,a):
        if i*i==a:
            break
    if i*i==a:
        break
    a+=1
print(a)
```


Input Format:

Integer input from stdin.

Output Format:

Perfect square greater than N.

Example Input:

10

Output:

16

Ex. No. : 4.5 **Date: 16/03/24**

Register No.: 230701348 **Name: N SUBRAMANIAN**

Nth Fibonacci

Write a program to return the nth number in the fibonacci series. The value of N will be passed to the program 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.

```
n =int(input())
a=0
b=1
c=0
for i in range(1,n-1):
        c=a+b
        a=b
        b=c
        n-=1
print(c)
```


For example:

Input:

7

Output

8

Ex. No. : 4.6 **Date: 16/03/24**

Register No.: 230701348 **Name: N SUBRAMANIAN**

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 to print number is Disarium or not.

```
a=int(input())
ori=a
t=a
co=0
while t>0:
        t//=10
        co+=1
sum=0
for i in range(co,0,-1):
        r=a%10
        p=r**i
        sum+=p
        a//=10
if sum==ori:
        print("Yes")
else:
        print("No")
```


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:

Input	Result
175	Yes
123	No

Ex. No. : 4.7 **Date: 16/03/24**

Register No.: 230701348 **Name: N SUBRAMANIAN**

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)

```
n=int(input())
a=1
sum=0
for i in range(n):
    sum+=a
    a=a*10+1
print(sum)
```


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

Ex. No. : 4.8 **Date: 16/03/24**

Register No.: 230701348 **Name: N SUBRAMANIAN**

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'.

```
a=int(input())
t=a
co=0
while t>0:
    t//=10
    co+=1
while a>0:
    d=a%10
    b=a//10
    c=0
    while b>0:
        e=b%10
        if d==e:
            c+=1
        b//=10
    if c>0:
        co--1
    a//-10
print(co)
```

For example:

Input	Result
292	2
1015	3

Ex. No. : 4.9 **Date: 16/03/24**

Register No.: 230701348 **Name: N SUBRAMANIAN**

Product of single digit

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

```
a=int(input())
flag=0
for i in range(1,a):
    if a%i==0:
        if a/i<10:
            flag=1
if flag:
    print("Yes")
else:
    print("No")
```


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

Ex. No. : 4.10 **Date: 16/03/24**

Register No.: 230701348 **Name: N SUBRAMANIAN**

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.

```
a=int(input())
flag=0
for i in range(1,a//2+1): if(i*i==a+1): flag=1
if flag: print("Yes")
else: print("No")
```


Input Format:

Single integer input.

Output Format:

Yes or No.

Example Input:

24

Output:

Yes

Example Input:

26

Output:

No

For example:

Input	Result
24	Yes