

## **04 - Iteration Control Structures**

**Ex. No. : 4.1**

**Date:**

**Register No.: 230701362**

**Name: THARUNRAJ I**

---

**Factors of a number**

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

**For example:**

Input	Result
20	1 2 4 5 10 20

**CODE:**

```
n=int(input())
```

```
i=1
```

```
while(i<=n):
```

```
    if(n%i==0):
```

```
        print(i,end=" ")
```

```
    i+=1
```

**Ex. No. : 4.2**

**Date:**

**Register No.: 230701362**

**Name: THARUNRAJ I**

---

### **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  $\geq 1$  and  $\leq 25000$ .

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.

**For example:**

<b>Inp ut</b>	<b>Result</b>
292	1
1015	2
108	3

Input	Result
22	0

CODE:

```

n=int(input())
l=[ ]
co=0
digi=0
while(n!=0):
    r=n%10
    n=n//10
    l.append(r)
for i in range(len(l)):
    co=0
    for j in range(len(l)):
        if(l[i]==l[j] and i!=j):
            co+=1
    if(co==0):
        digi+=1
print(digi)

```

**Ex. No. : 4.3**

**Date:**

**Register No.: 230701362**

**Name: THARUNRAJ I**

---

### **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:  $2 \leq N \leq 5000$ , 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:**

<b>Inpu t</b>	<b>Resu lt</b>
7	2
10	1

CODE:

```
n=int(input())
f=0
for i in range(2,n):
    if(n%i==0):
        f=1
        break
if(f==0):
    print(2)
else:
    print(1)
```

**Ex. No. : 4.4**

**Date:**

**Register No.:230701362**

**Name: THARUNRAJ I**

---

### **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

**CODE:**

```
n=int(input())
```

```
i=1
```

```
while(n>i*i):
```

```
    i+=1
```

```
print(i*i)
```

**Ex. No. : 4.5**

**Date:**

**Register No.: 230701362**

**Name: THARUNRAJ I**

---

### **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.

**For example:**

**Input:**

7

**Output**

8

CODE:

```
n=int(input())
```

```
a=0
```

```
b=1
```

```
c=1
```

```
i=0
```

```
while(i<n-2):
```

```
    c=a+b
```

```
    a=b
```

```
    b=c
```

```
    i+=1
```

```
if(n==1):
```

```
    print(0)
```

```
else:
```

```
    print(c)
```



**Ex. No. : 4.6**

**Date:**

**Register No.: 230701362**

**Name: THARUNRAJ I**

---

### **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.

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
-----	----

**CODE:**

```
n=int(input())
d=n
digi=0
sum=0
while(d!=0):
    d=d//10
    digi+=1
d=n
while(d!=0):
    r=d%10
    sum=r*digi+sum
    digi-=1
    d=d//10
if(sum==n):
    print("Yes")
else:
    print("No")
```

**Ex. No. : 4.7**

**Date:**

**Register No.: 230701362**

**Name: THARUNRAJ I**

---

### **Sum of Series**

Write a program 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

Explanation:

as input is 4, have to take 4 terms.

$1 + 11 + 111 + 1111$

Test Case 2

Input

6

Output

123456

**For example:**

<b>Inpu t</b>	<b>Resu lt</b>
3	123

**CODE:**

```
n=int(input())
```

```
sum=0
```

```
var=0
```

```
for i in range(n):
```

```
    var=var*10+1
```

```
    sum=sum+var
```

```
print(sum)
```

**Ex. No. : 4.8**

**Date:**

**Register No.: 230701362**

**Name: THARUNRAJ I**

---

### **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  $\geq 1$  and  $\leq 25000$ .

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:**

<b>Input</b>	<b>Result</b>
292	2
1015	3

**CODE:**

```
n=int(input())
l=[ ]

while(n!=0):

    d=n%10

    if d not in l:

        l.append(d)

    n=n//10

print(len(l))
```

**Ex. No. : 4.9**

**Date:**

**Register No.: 230701362**

**Name: THARUNRAJ I**

---

**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

**CODE:**

```
n=int(input())
```

```
i=2
```

```
while(n%i!=0 and i<9 and n//i<9):
```

```
    i+=1
```

```
if(i==9):
```

```
    print("No")
```

```
else:
```

```
    print("Yes")
```



**Ex. No. : 4.10**

**Date:**

**Register No.: 230701362**

**Name: THARUNRAJ I**

---

**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:**

<b>Inp ut</b>	<b>Resu lt</b>
24	Yes

CODE:

```
n=int(input())
```

```
i=0
```

```
while(n>i*i):
```

```
    i+=1
```

```
if(n+1==i*i):
```

```
    print("Yes")
```

```
else:
```

```
    print("No")
```

