

Program:

```
a=int(input())
```

```
t=1
```

```
s=0
```

```
for i in range(a)
```

```
    s+=t
```

```
    t=t*10+1
```

```
print(s)
```

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

Ex. No. : 4.8

Date: 13.04.24

Register No.: 231901018

Name: Kavın Sainath S

Prime Checking

Write a program that finds whether the given number 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.

Example 1: if the given number is 7, the method must return 2

Example 2: if the given number 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")
```

	Input	Expected	Got	
✓	7	2	2	✓
✓	10	1	1	✓

Ex. No. : 4.9

Date: 13.04.24

Register No.: 231901018

Name: Kavin Sainath S

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

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:
        print("No")
```

	Input	Expected	Got	
✓	175	Yes	Yes	✓
✓	123	No	No	✓

Ex. No. : 4.10

Date: 13.04.24

Register No.: 231901018

Name: Kavin Sainath S

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	Result
24	Yes

Program:

```
import math  
  
a=int(input())  
  
b=a+1  
  
c=math.sqrt(b)  
  
if(c==int(c)):  
    print("Yes")  
  
else:  
    print("No")
```

	Input	Expected	Got	
✓	24	Yes	Yes	✓
✓	26	No	No	✓

05 - Strings in Python

String characters balance Test

Write a program to check if two strings are balanced. For example, strings `1` and `2` are balanced if all the characters in the `1` are present in `2`. The character's position doesn't matter. If balanced display as `"true"`, otherwise `"false"`.

Input	Result
Yn	True
PYnative	

For example:

Program:

```
a=input()
b=input()
if a in b or b in a:
    print("True")
else:
    print("False")
```

	Input	Expected	Got	
✓	Yn PYnative	True	True	✓
✓	Ynf PYnative	False	False	✓

Ex. No. : 5.2

Date: 17.04.24

Register No.: 231901018

Name: Kavin Sainath S

Decompress the String

Assumethatthegivenstringhasenoughmemory.Don'tuseanyextraspacespace(IN-PLACE)

SampleInput1

a2b4c6

SampleOutput1

aabbbbcccccc

Program:

```
s=input()
r=""
i=0
while i< len(s):
    char=s[i]
    i+=1
    num=""
    while i<len(s) and s[i].isdigit():
        num+=s[i]
        i+=1
    r+=char*int(num)
print(r)
```

	Input	Expected	Got	
✓	a2b4c6	aabbbbcccccc	aabbbbcccccc	✓
✓	a12b3d4	aaaaaaaaaabbddddd	aaaaaaaaaabbddddd	✓

Ex. No. : 5.3

Date: 17.04.24

Register No.: 231901018

Name: Kavin Sainath S

First N Common Chars

Two string values S1, S2 are passed as the input. The program must print first N characters present in S1 which are also present in S2.

Input Format:

The first line contains S1.

The second line contains S2.

The third line contains N.

Output Format:

The first line contains the N characters present in S1 which are also present in S2.

Boundary Conditions:

$2 \leq N \leq 10$

$2 \leq \text{Length of } S1, S2 \leq 1000$

Example Input/Output 1:

Input:

```
abcbde
cdefghbb
3
```

Output:

```
bcd
```

Note:

b occurs twice in common but must be printed only once.

Program:

```
a=input()
b=input()
n=int(input())
bset=set(b)
cc=[]
c=0
for i in a:
    if i in bset and i not in cc:
        cc.append(i)
        c=c+1
    if(c==n):
        break
s="".join(cc)
print(s)
```

	Input	Expected	Got	
✓	abcbde cdefghbb 3	bcd	bcd	✓

Ex. No. : 5.4

Date: 17.04.24

Register No.: 231901018

Name: Kavin Sainath S

Username Domain Extension

Given a string `S` which is of the format `USERNAME@DOMAIN.EXTENSION`, the program must print the `EXTENSION`, `DOMAIN`, `USERNAME` in the reverse order.

Input Format

The first line contains `S`.

Output Format

The first line contains `EXTENSION`.

The second line contains `DOMAIN`.

The third line contains `USERNAME`.

Boundary Condition:

$1 \leq \text{Length of } S \leq 100$

Example Input/Output 1:

Input

`vijayakumar.r@rajalakshmi.edu.in`

Output

`edu.in`

`rajalakshmi`

`vijayakumar.r`

Program:

```
s=input()
at=s.index('@')
dot=s.index('.')
username=s[:at]
domain=s[at+1:dot]
exten=s[dot+1:]
print(exten)
print(domain)
print(username)
```

	Input	Expected	Got	
✓	abcd@gmail.com	com gmail abcd	com gmail abcd	✓

Ex. No. : 5.5

Date: 17.04.24