vl dharshini
231501038
aiml a

WEEK 4

1) 1) 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.

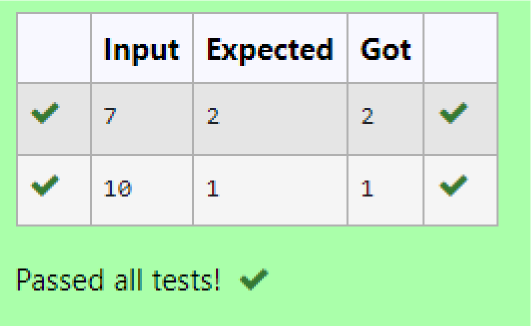
Example1: if the given number N is 7, the method must return 2

Example2: if the given number N is 10, the method must return 1

PROGRAM:

```
a= int(input())
if a % 2 == 1:
   print("2")
else:
   print("1")
```

OUTPUT:



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

Passed all tests! ✓

2) 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.

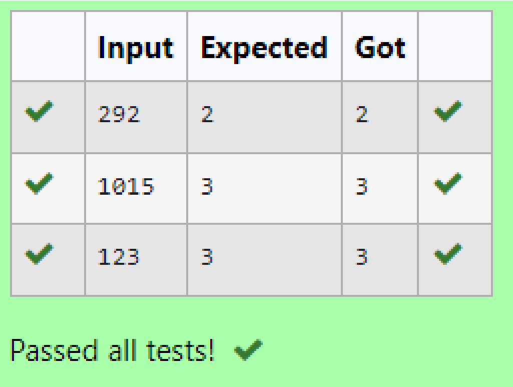
If the given number is 292, the program should return 2 because there are only 2 unique digits '2' and '9' in this number

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

PROGRAM:

```
print(len(set(input())))
```

OUTPUT:



	Input	Expected	Got	
✓	292	2	2	✓
✓	1015	3	3	✓
✓	123	3	3	✓

Passed all tests! ✓

3) 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

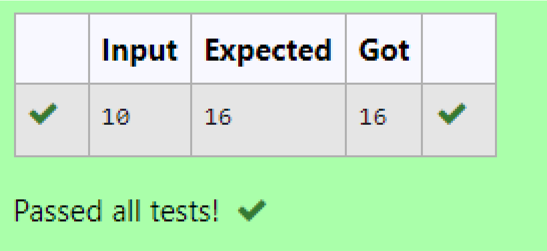
Output:

16

PROGRAM:

```
import math as m
a = int(input())
for i in range(a, 10000):
        X = m.sqrt(i)
        if(i % x == 0):
          print(i)
          break
```

OUTPUT:



	Input	Expected	Got	
✓	10	16	16	✓

Passed all tests! ✓

4) 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'.

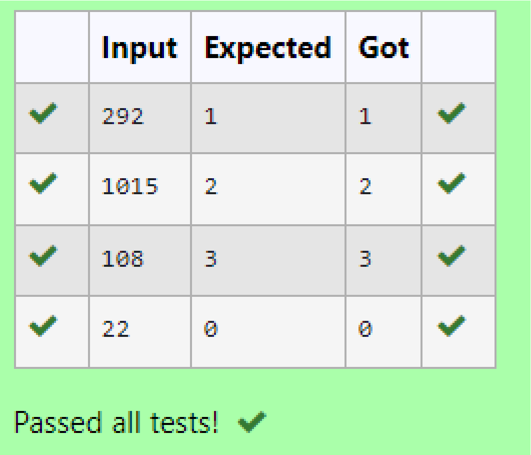
If the given number is 108, the program should return 3 because there are 3 non--repeated digits in this number, '1', '0', and '8'.

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

PROGRAM:

```
n = (input())
l=[0] *10
for i in n:
    l[int(i)]+=1
    c=0
for i in l:
    if i==1:
    c += 1
print(c)
```

OUTPUT:



	Input	Expected	Got	
✓	292	1	1	✓
✓	1015	2	2	✓
✓	108	3	3	✓
✓	22	0	0	✓

Passed all tests! ✓

5) 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

WEEK 4

Test Case 1

Input

4

Output

1234

Test Case 2

Input

6

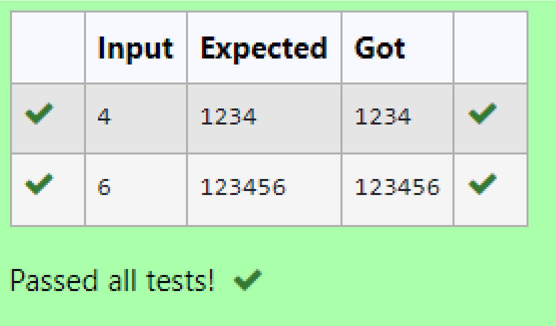
Output

123456

PROGRAM:

```
a=int(input())

n=0

t=0

for i in range (a):

    n=n*10+1

    t+=n

print(t)
```

OUTPUT:



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

Passed all tests! ✓

6) 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

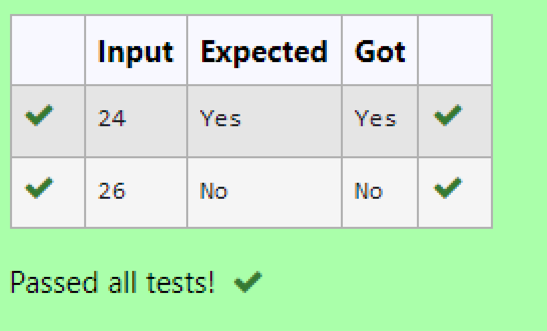
Output:

No

PROGRAM:

```
Import math as b
a=int(input())
a=a+1
m=b.sqrt(a)
if m %2==0 or m%2==1:
    print("Yes")
else:
    print("No")
```

OUTPUT:



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

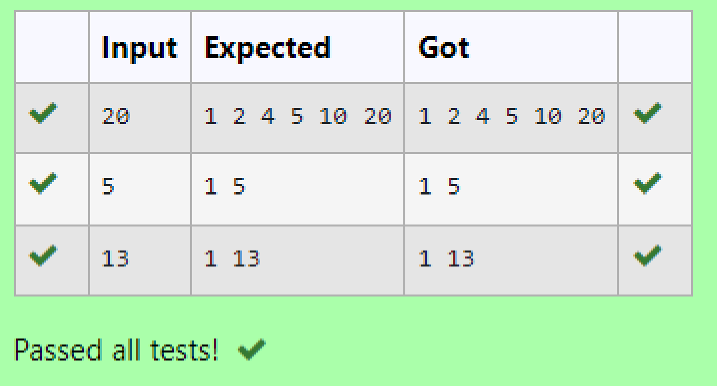
Passed all tests! ✓

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

PROGRAM:

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

OUTPUT:



	Input	Expected	Got	
✓	20	1 2 4 5 10 20	1 2 4 5 10 20	✓
✓	5	1 5	1 5	✓
✓	13	1 13	1 13	✓

Passed all tests! ✓

8) 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.

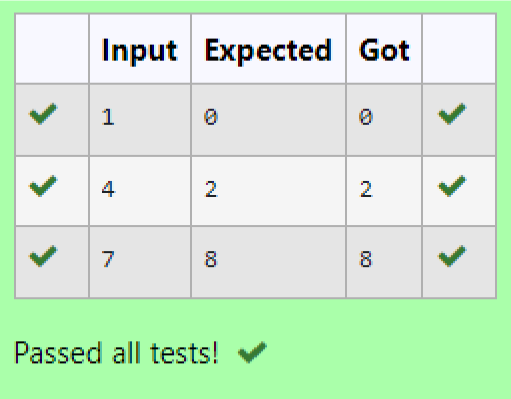
- first Fibonacci number is 0,
- second Fibonacci number is 1,
- third Fibonacci number is 1,
- fourth Fibonacci number is 2,
- fifth Fibonacci number is 3,
- sixth Fibonacci number is 5,
- seventh Fibonacci number is 8, and so on.

PROGRAM:

```
z=int(input())
```

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

OUTPUT:



	Input	Expected	Got	
✓	1	0	0	✓
✓	4	2	2	✓
✓	7	8	8	✓

Passed all tests! ✓

9) 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

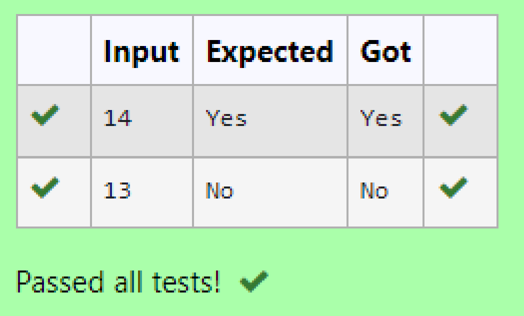
Output:

No

PROGRAM:

```
a=int(input())
if a%2==0 or %3==0 or a%5==0 ora%6==0:
        print(“Yes”)
else:
        print(“No”)
```

OUTPUT:



	Input	Expected	Got	
✓	14	Yes	Yes	✓
✓	13	No	No	✓

Passed all tests! ✓

10) 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

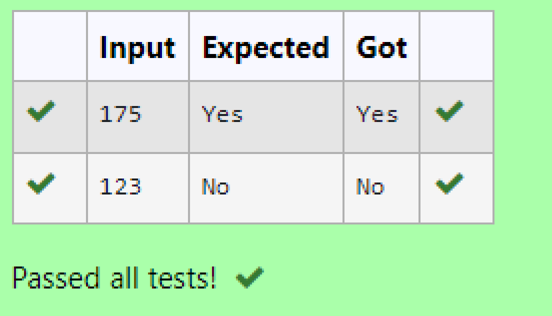
Output:

No

PROGRAM:

```
a=int(input())

x=str(a)

count=0

for i in range (len(x)):

    tot=pow(int(x[i]),i=1)

    count +=a:

if count ==a:

    print(“Yes”)

else:

    print(“No”)
```

OUTPUT:



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

Passed all tests! ✓