1. 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

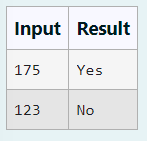
Example Input:

123

Output:

No

For example:



Answer

k=input()

s=1

ans=0

for i in k:

ans+=int(i)\*\*s

s+=1

if(ans==int(k)):

print("Yes")

else:

print("No")

2. 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

Answer

a=abs(int(input()))

b=a

for i in range(2,10):

while(b%i==0):

b/=i

if b==1:

print("Yes")

else:

print("No")

3.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

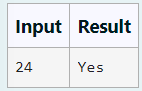
Example Input:

26

Output:

No

For example:



Answer

n=int(input())

n+=1

sn=int(n\*\*0.5)

if sn\*\*2==n:

print("Yes")

else:

print("No")

4.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,

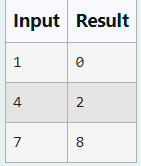
• fourth Fibonacci number is 2,

• fifth Fibonacci number is 3,

• sixth Fibonacci number is 5,

• seventh Fibonacci number is 8, and so on.

For example:



Answer

a=0

b=1

n=int(input())

for i in range(n-1):

a,b=b,a+b

print(a)

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

Test Case 1

Input

4

Output

1234

Test Case 2

Input

6

Output

123456

Answer

n=int(input())

b=((10\*(10\*\*n-1)/9)-n)/9

print(int(b))

6.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

Answer

n=int(input())

print((int(n\*\*0.5)+1)\*\*2)

7.Write a program that finds whether the given number N is Prime or not.

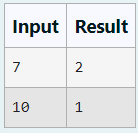
If the number is prime, the program should return 2 else it must return 1.

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:



Answer

a=int(input())

flag=1

for i in range(2,int(a\*\*0.5)+1):

if(a%i==0):

flag=0

if(flag):

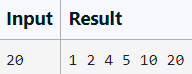
print(2)

else:

print(1)

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

For example:



Answer

n=int(input())

factors=[]

for i in range(1,n+1):

if n % i==0:

factors.append(i)

print(\*factors)

9.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.

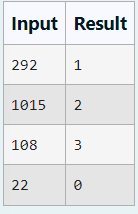
If the given number is 292, the program should return 1 because there is only 1 non-­repeated digit '9' in this number

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.

For example:



Answer

arr=[0,0,0,0,0,0,0,0,0,0]

a=input()

count=0

for i in a:

arr[int(i)]+=1

for j in arr:

if j==1:

count+=1

print(count)

10.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.

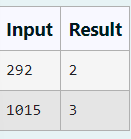
Assumption: The input number will be a positive integer number >= 1 and <= 25000.

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:



Answer

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