Ex. No. : 4.1

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

n=int(input())

x=[]

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

    if n%i == 0:

        x.append(i)

        x.append(int(n/i))

factors = list(set(x))

x.sort()

for i in range(len(x)):

    print(x[i],end =' ')

Ex. No. : 4.2

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.

For example:

Input

Result

292

1

1015

2

108

3

22

0

n=int(input())

def func(n):

    strn = str(n)

    dig = {}

    nrc = 0

    for i in strn:

        if i in dig:

            dig[i] += 1

        else:

            dig[i]=1

    for i,cnt in dig.items():

        if cnt == 1:

            nrc += 1

    return nrc

print(func(n))

Ex. No. : 4.3

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.

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

n=int(input())

count=0

if(n>=2 and n<=5000):

    for i in range(2,n//2):

        if(n%i==0):

            print(1)

            count+=1

            break

if(count==0):

    print(2)

Ex. No. : 4.4

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

n=int(input())

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

Ex. No. : 4.5

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

n=int(input())

if(n==1):

    print(0)

elif(n==2):

    print(1)

else:

     a=0

     b=1

     for i in range(n-2):

         temp=a+b

         a=b

         b=temp

     print(b)

Ex. No. : 4.6

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

n=int(input())

def check(n):

    cnt = len(str(n))

    s = 0

    x=n

    while(x != 0):

        r = x%10

        s = (int(s+(r\*\*cnt)))

        cnt -= 1

        x = x//10

    if s == n:

        return 1

    else:

        return 0

if(check(n) == 1):

    print("Yes")

else:

    print("No")

Ex. No. : 4.7

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)

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

n=int(input())

temp=0

for i in range(n):

    temp=temp\*10+(i+1)

print(temp)

Ex. No. : 4.8

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'.

For example:

Input

Result

292

2

1015

3

a=int(input())

n=str(a)

l=len(str(n))

count=0

s=set()

for i in range(0,l):

    if(n[i] not in s):

        count+=1

        s.add(n[i])

print(count)

Ex. No. : 4.9

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

n=int(input())

l=[2,3,4,5,6,7,8,9]

x=False

for i in l:

    if n%i==0:

        x=True

        break

if x:

    print("Yes")

else:

    print("No")

Ex. No. : 4.10

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

n=int(input())

k=(n+1)//2

count=0

for i in range(0,k+1):

    if(i\*i==n+1):

        count+=1

        break

if(count==0):

    print("No")

else:

    print("Yes"