Problem 1 - Print the following pattern. Write a program to use for loop to print the following reverse number pattern.

```
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
```

```
# Code here
n = int(input("Enter the number: "))
for i in range(n,0,-1):
  for j in range(i,0,-1):
    print(j,end=" ")
  print()
```

```
Enter the number: 5
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
```

+ Code + Text

```
n = int(input(" "))
for i in range(1,n+1):
  for j in range(1,i+1):
    print(j,end = " ")
  print()
```

```
 5
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

Problem 2 : Print the following pattern.

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

```
# Code here
n = int(input(" "))
for i in range(1,n+1):
  for j in range(1,i+1):
    print("*",end=" ")
  print()
for i in range(n,1,-1):
  for j in range(i,1,-1):
    print("*",end=" ")
  print()
```

```
 5
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
```

```
    * *
    *
```

Problem 3 :Write a program to pring the following pattern

```
    *
   * * *
 * * * * *
```

```
# Code here
n = int(input(""))
for i in range(1,n+1):
  print("  " * (n - i), end="")
  for j in range(0,(2*i) -1):
    print("*",end=" ")
  print()
```

```
5
          *
        * * *
      * * * * *
    * * * * * * *
  * * * * * * * * *
```

Problem 4 :Write a program to print the following pattern

1

2 1

3 2 1

4 3 2 1

5 4 3 2 1

```
# Code here
n = int(input(""))
for i in range(1,n+1):
  for j in range(i,0,-1):
    print(j,end=" ")
  print()
```

```
5
1
2 1
3 2 1
4 3 2 1
5 4 3 2 1
```

Problem 5 : Write a Python Program to Find the Sum of the Series till the nth term:

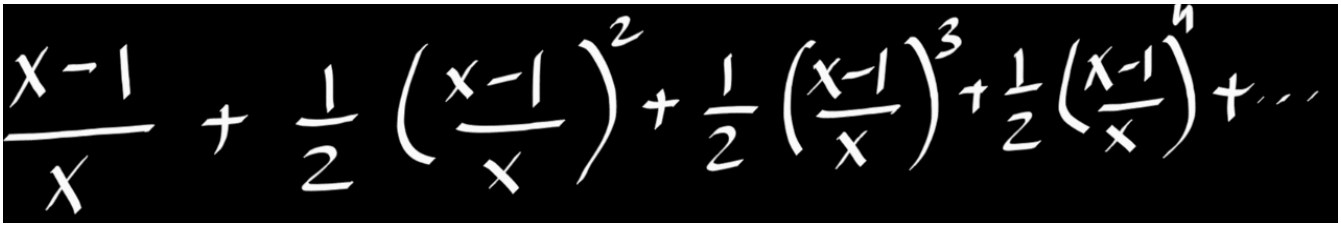
1 + x^2/2 + x^3/3 + … x^n/n

n will be provided by the user

```
# Code here
nth = int(input("Enter the nth term: "))
x = int(input("Enter the value of x: "))
i = 2
sum = 1
while i <= nth:
  sum = sum + (x**i)/i
  i = i + 1
print(sum)
```

```
Enter the nth term: 4
Enter the value of x: 2
9.666666666666666
```

Problem 6 : The natural logarithm can be approximated by the following series.



If x is input through the keyboard, write a program to calculate the sum of the first seven terms of this series.

```
# Code here
x = float(input("Enter the value of x (|x| < 1): "))

sum = 0
power = 1

for i in range(1, 14, 2):  # i = 1, 3, 5, ..., 13 (7 terms)
    term = (x ** i) / i
    sum += term

ln_result = 2 * sum
print("Approximate ln((1 + x)/(1 - x)):", ln_result)
```

```
Enter the value of x (|x| < 1): 0.5
Approximate ln((1 + x)/(1 - x)): 1.0986070624254214
```

Problem 7 - Find the sum of the series upto n terms.

Write a program to calculate the sum of series up to n term. For example, if n =5 the series will become 2 + 22 + 222 + 2222 + 22222 = 24690. Take the user input and then calculate. And the output style should match which is given in the example.

Example 1:

```
Input:

 5

Output:

 2+22+222+2222+22222
 Sum of above series is: 24690
```

```
# Code here
inpt = int(input("Enter number: "))
term = 0
sum = 0
i = 1
sum = 0
while i <= inpt:
  term = term * 10 + 2
  sum = sum + term
  print(term,end=" + ")
  i = i + 1
print(sum)
```

```
Enter number: 5
2 + 22 + 222 + 2222 + 22222 + 24690
```

Problem 8 : Write a program to print all the unique combinations of 1,2,3 and 4

```
Output :
```

```
1 2 3 4
1 2 4 3
1 3 2 4
1 3 4 2
1 4 2 3
1 4 3 2
2 1 3 4
2 1 4 3
2 3 1 4
2 3 4 1
2 4 1 3
.
.
and so on
```

```
# Code here
for i in range(1,5):
  for j in range(1,5):
    for k in range(1,5):
      for l in range(1,5):
        if i != j and i!=k and i!=l and j!=k and j!=l and k!= l:
          print(i,j,k,l)
      #print()
    #print()
  #print()
```

```
1 2 3 4
1 2 4 3
1 3 2 4
1 3 4 2
1 4 2 3
1 4 3 2
2 1 3 4
2 1 4 3
2 3 1 4
2 3 4 1
2 4 1 3
2 4 3 1
3 1 2 4
3 1 4 2
3 2 1 4
3 2 4 1
3 4 1 2
3 4 2 1
4 1 2 3
4 1 3 2
4 2 1 3
4 2 3 1
4 3 1 2
4 3 2 1
```

Problem 9 : Write a program that will take a decimal number as input and prints out the binary equivalent of the number

```
# Code to convert decimal to binary
num = int(input("Enter the number: "))

# Edge case: if the number is 0
if num == 0:
    print("Binary equivalent: 0")
else:
 binary = ""
 while num > 0:
    binary = str(num % 2) + binary
    num = num // 2

print("Binary equivalent:", binary)
```

```
Enter the number: 12
Binary equivalent: 1100
```

Problem 10: Write a program that will take 2 numbers as input and prints the LCM and HCF of those 2 numbers

```
# Code here
```

Problem 11: Create Short Form from initial character

Given a string create short form ofthe string from Initial character. Short form should be capitalised.

Example:

```
Input:

 Data science mentorship program

Output:

 DSMP
```

```
lst = ["data","mana","kaka","laka"]
lst[0][0]
```

```
'd'
```

```
print("a"+"b"+"c")
```

```
abc
```

```
# Code here
str = "Data science mentorship program"
lst = str.split()
i = 0
new_str = ""
while i < len(lst):
    new_str = new_str + lst[i][0].upper()
    i = i + 1
print(new_str)
```

```
DSMP
```

Problem 12: Append second string in the middle of first string

```
Input:

 campusx
 data

Output:

 camdatapusx
```

```
# Code here
word_1 = "CampusX"
word_2 = "data"
i = 0
before = word_1[0:3]
after = word_1[3:]
new = before + word_2 + after
print(new)
```

```
CamdatapusX
```

Problem 13 :Given string contains a combination of the lower and upper case letters. Write a program to arrange the characters of a string so that all lowercase letters should come first.

Given:

```
str1 = PyNaTive
```

Expected Output:

```
yaivePNT
```

```
# Code here
str1 = "PyNaTive"
l = ""
u = ""
for i in str1:
  if i.islower():
   l = l + i
  else:
   u = u + i
print(l+u)
```

```
yaivePNT
```

Problem 14: Take a alphanumeric string input and print the sum and average of the digits that appear in the string, ignoring all other characters.

Input:

```
hel123O4every093
```

Output:

```
 Sum: 22
 Avg: 2.75
```

```
# Code here
x = "hel123O4every093"
n = 0
c = ""
count = 1
for i in x:
  if i.isnumeric():
    i = int(i)
    n = n + i
    count = count + 1
  else:
    c = c + i
print(n)
print(n/count)
```

```
22
 2.75
```

Problem 15: Removal of all characters from a string except integers

```
Given:

 str1 = 'I am 25 years and 10 months old'

Expected Output:

 2510
```

```
# Code here
str1 = 'I am 25 years and 10 months old'
```

```
char = ""
for i in str1:
  if i.isnumeric():
    char = char + i
print(char)
```

```
2510
```

Problem 16 : Check whether the string is Symmetrical.

Statement: Given a string. the task is to check if the string is symmetrical or not. A string is said to be symmetrical if both the halves of the string are the same.

Example 1:

```
Input

 khokho

Output

 The entered string is symmetrical
```

```
# Code here
x = "khokhi"
i = 0
mid = int(len(x)/2)
if x[:mid] == x[mid:]:
  print("The entered string is symmetrical")
else:
  print("Not symmetrical")
```

```
Not symmetrical
```

Problem 17 : Reverse words in a given String

Statement: We are given a string and we need to reverse words of a given string.

Example 1:

```
Input:

 geeks quiz practice code

Output:

 code practice quiz geeks
```

Example 2:

```
Input:

 my name is laxmi

Output:

 laxmi is name my
```

```
# Code here
x = input("Enter the sentence: ")
word = x.split()
reverse_string = ""
for i in range((len(word)-1),-1,-1):
  reverse_string = reverse_string + word[i] + " "
print(reverse_string)
```

```
Enter the sentence: laxmi is name my
my name is laxmi
```

Problem 18 : Find uncommon words from two Strings.

Statement: Given two sentences as strings **A** and **B**. The task is to return a list of all uncommon words. A word is uncommon if it appears exactly once in any one of the sentences, and does not appear in the other sentence. Note: A sentence is a string of space-separated words. Each word consists only of lowercase letters.

Example 1:

```
Input:

A = "apple banana mango"
B = "banana fruits mango"

Output:

['apple', 'fruits']
```

```
# Code here
A = "apple banana mango"
B = "banana fruits mango"
new = A + " " + B
word = new.split()

new_list = []

for i in word:
    if word.count(i) == 1 and i not in new_list:
        new_list.append(i)

print(new_list)
```

```
['apple', 'fruits']
```

Problem 19 : Word location in String.

Statement: Find a location of a word in a given sentence.

Example 1:

```
Input:

Sentence: We can learn data science through campusx mentorship program.

word: campusx

Output:

Location of the word is 7.
```

Note- Don't use index/find functions

```
# Code here
sentence = input("Enter the sentence: ")
find_word = input("Enter the word you want to find: ")
words = sentence.split()
for i in range(len(words)):
  if words[i] == find_word:
        print(f"Word found at position {i}, with length {len(find_word)}")
        break
else:
    print("Word not found")
```

```
Enter the sentence: We can learn data science through campusx mentorship program
Enter the word you want to find: campusx
```

```
Word found at position 6, with length 7
```

```
sentence = input("Enter the sentence: ")
find_word = input("Enter the word youy want to find ")
```

Problem 20 : Write a program that can remove all the duplicate characters from a string. User will provide the input.

```
# Code here
enter = input("Enter the word: ")
i = 0
new = ""
while i < len(enter):
  if enter[i] not in new:
    new = new + enter[i]
  i = i + 1
print(new)
```

```
Enter the word: programming
progamin
```