`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

rows = int(input('enter the rows'))

for i in range(0,rows):
  for j in range(rows-i,0,-1):
    print(j,end=' ')
  print()
```

```
enter the rows5
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
```

`Problem 2`: Print the following pattern.

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

```
# Code here
rows = int(input('enter the rows'))

for i in range(1,rows+1):
  for j in range(0,i):
    print('*',end=' ')
  print()

for i in range(1,rows):
  for j in range(rows-i,0,-1):
    print('*',end=' ')
  print()
```

```
enter the rows5
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

`Problem 3`:Write a program to pring the following pattern

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

```
# Code here
rows = 6
for i in range(1,rows+1):
  print(' '*rows,end='')
  print('* '*i)
  rows = rows - 1
```

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

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
rows = 5

for i in range(1,rows+1):
  for j in range(i,0,-1):
    print(j,end=' ')
  print()
```

```
    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

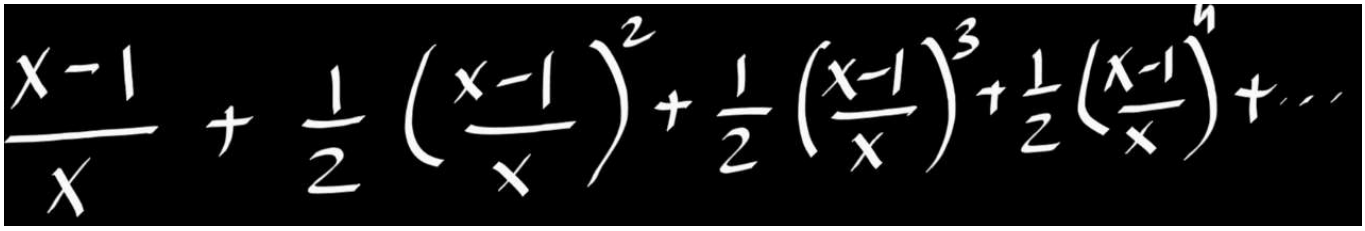
n will be provided by the user

```
# Code here
x = 10
n = 5

sum = 1
s = ''
print('1 + ',end='')
for i in range(2,n+1):
  sum = sum + x**i/i
  s = s + 'x^{}/{} +'.format(i,i)
print(s[:-1])
print(sum)
```

```
    1 + x^2/2 +x^3/3 +x^4/4 +x^5/5
    22884.333333333332
```

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 = 10
n = 5

sum = 0
s = ''
for i in range(1,n+1):
  sum = sum + (1/i)*((x-1)/x)**i
  s = s + '(1/{})*((x-1)/x)^{} +'.format(i,i)
print(s[:-1])
print(sum)
```

```
(1/1)*((x-1)/x)^1 +(1/2)*((x-1)/x)^2 +(1/3)*((x-1)/x)^3 +(1/4)*((x-1)/x)^4 +(1/5)*((x-1)/x)^5
1.8301230000000004
```

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
n = int(input('enter the number of terms'))

start = 2
sum = 0

for i in range(0,n):
  if i < n-1:
    print(start,end='+')
  else:
    print(start)

  sum = sum + start
  start = start*10 + 2

print(sum)
```

```
enter the number of terms5
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 m in range(1,5):
        print(i,j,k,m)
```

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

```
# Code here

n = int(input('enter the number'))
binary = []
while n > 0:

  binary.append(n%2)
  n = n//2

for i in binary[::-1]:
  print(i,end='')
```

```
enter the number3
11
```

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

```
# Code here
x = int(input('enter 1st number'))
y = int(input('enter 2nd number'))

if x>y:
  greater = x
else:
  greater = y

while True:
  if (greater % x == 0) and (greater % y == 0):
    lcm = greater
    break

  greater = greater + 1

print(lcm)
```

```
enter 1st number10
enter 2nd number15
30
```

```
# Code here
x = int(input('enter 1st number'))
y = int(input('enter 2nd number'))

if x<y:
  smaller = x
else:
  smaller = y

for i in range(1,smaller+1):
  if (x % i == 0) and (y % i == 0):
    hcf = i

print(hcf)
```

```
enter 1st number12
enter 2nd number16
4
```

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
```

```
# Code here

inp = 'Data science Mentorship Program'

res = ''

for i in inp.split():
  res = res + i[0].upper()

print(res)
```

```
DSMP
```

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

```
Input:

 campusx
 data

Output:

 camdatapusx
```

```
# Code here

s1 = input('enter the 1st string')
s2 = input('enter the 2nd string')

print(s1[0:int(len(s1)/2)] + s2 + s1[int(len(s1)/2):])
```

```
enter the 1st stringwill
enter the 2nd stringdo
widoll
```

`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

s = 'PyNaTive'

upper = ''
lower = ''

for i in s:
  if i.islower():
    lower = lower + i
  else:
    upper = upper + i

print(lower + upper)
```

```
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
s = 'hel123O4every093'

sum = 0
count = 0

for i in s:
  if i.isdigit():
    sum = sum + int(i)
    count += 1

print(sum)
print(sum/count)
print(count)
```

```
22
3.142857142857143
7
```

`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
s = 'I am 25 years and 10 months old'

res = ''

for i in s:
  if i.isdigit():
    res = res + i

print(res)
```

```
 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
s = input('enter the string')

if len(s)%2 == 0:
  s1 = s[0:len(s)//2]
  s2 = s[len(s)//2:]
else:
  s1 = s[0:len(s)//2]
  s2 = s[len(s)//2 + 1:]

if s1 == s2:
  print('symmetrical')
else:
  print('not symmetrical')
```

```
 enter the stringmadam
 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
s = 'code practice quiz geeks'

L = []

for i in s.split():
  L.append(i)

L = L[::-1]
print(" ".join(L))
```

```
     geeks quiz practice code
```

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"

L = []

for i in A.split():
  if i not in B and i not in L:
    L.append(i)

for j in B.split():
  if j not in A and j not in L:
    L.append(j)

print(L)
```

```
     ['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
s = 'We can learn data science through campusx mentorship program.'
word = 'campusx'

pos = 0

for i in s.split():
  pos += 1
  if i == word:
    break

print(pos)
```

```
 7
```

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

```
# Code here
s = 'aaaabbbbbccccdddeeeeffff'

res = ''

for i in s:
  if i not in res:
    res = res + i

print(res)
```

```
 abcdef
```

Start coding or generate with AI.