Programming with Python
Note: Brief Summary of contents discussed. (Feb 3,4-2025)

While loop: Used for repeatedly executing a sequence of statements based on some condition being True.

```
while <test condition>:
    <Statements>
```

Exercise (use while loop):

- Write a program to compute the factorial of a given number using a while loop.
- Write a program to find the sum of digits of a given number using while loop (and // and % operator)

break and continue statement

The `break` statement is used for exiting from the immediate loop to the statement following the body of the loop.

The `continue` statement is used to transfer the control to the next iteration of the loop without performing any action.

Example:
Average temperature in a city based on readings received from different sensors.

```
avg_tmp = 0
count = 1
while True:
    temp = input("Enter temperature readings: ")
    if temp == '':
        break;           #End of input
    temp = eval(temp)
    if temp < 0 or temp > 45:
        print('Invalid temperature')
        continue         #Enter temperature again
    count += 1
    avg_tmp = (avg_temp * (count - 1) + temp) / count

print("Average Temperature: ", avg_temp)
```

Give output:

a.

```
total = 0
count = 20
while count > 5:
    total += count
    count -= 1
print(total)
```

b.

```
total = 0
N = 4
for i in range(1, N+1):
    for j in range(1, i+1):
        total += j
print(total)
```

c.

```
total = 0
N = 5
for i in range(1, N+1):
    for j in range(1, N+1):
        total += 1
print(total)
```

Exercises (Try to write the program using while loop as well as for loop)

1. Write a Python program to input a number n from the user and print the reverse counting from n to 1.
2. Write a program to print all terms in Fibonacci Series less than a given number n.
3. Write a program to check if a number is prime or not.
   Note: use a variable flag = True, set it False and break as soon as you find a divisor
4. Write a program that uses two numbers as input parameters and returns their greatest common divisor (GCD, HCF).
5. Write a program that takes two numbers as input parameters and print their least common multiple.
   Note: Use while/if-else statements

6. Write a program to check if a given number is palindrome or not. (Note: Reverse the number using while loop and compare)

7. Write a program to approximate the square root of a given number using Newton's method (upto a given precision)

   $root_{n+1}$ iteration = 0.5 * ($root_n$ + num/($root_n$))

   $root_n$ is the root at the `n`th iteration

   $root_{n+1}$ is the root at the `n+1`th iteration

8. Write a program to convert a given positive decimal number to a binary number.

9. Write a Program that takes two numbers as input parameters and prints True or False depending on whether they are co-primes. Two numbers are said to be co-prime if they do not have any common divisor other than one.

10. (*We will develop this program further in subsequent classes*). A robot is placed at position (0,0) on a 2-dimensional grid. The robot can move **up, down, left, or right** based on user input. The goal is to allow the user to control the robot's movement using commands and display the final position after a series of moves. Write a program to implement the above with following rules using while/if-else.

    Rules:

a. The robot starts at (0,0).
b. The user inputs a series of moves:
    - "U" (Up) → Increases the y-coordinate by 1
    - "D" (Down) → Decreases the y-coordinate by 1
    - "L" (Left) → Decreases the x-coordinate by 1
    - "R" (Right) → Increases the x-coordinate by 1
c. The program should continue taking inputs until the user enters "STOP".
d. After the loop ends, display the final position of the robot.
e. (Optional) Update the above program to include the following condition.
    - Any move that makes the x or y-coordinate (-) negative is invalid and results in no move.