WEEK 2 - ASSIGNMENT 4
INTERMEDIATE WHILE LOOP

NOTE:

- **No need to submit anywhere, just keep track of all the PDF you made in a specific folder.**
- **Compare your solution with the solution I'll provide, in case of doubts, kindly reach out to me.**
- **You may get assignment solution in format of PDF or VIDEO solution, depending on the difficulty level.**

Q1. Create a function named **divs,** which will take two parameters **n1** and **n2**. Return the count of how many numbers from **1 to n1** are divisible by **n2.**

Q2. From 1 to 2000, print all the **LEAP YEARS,** using WHILE loop.

Q3. Create a function named **factorial** which takes a integer as an input and **returns** the factorial of that number.

Factorial of 5 means 5 x 4 x 3 x 2 x 1 = **120**

Q4. Create a function named **pattern** which takes an integer as an input and print the following pattern.

```
# Example 1
pattern(4)

# Output
10 20 30 40

# Example 2
pattern(11)

# Output
10 20 30 40 50 60 70 80 90 100 110
```

Q5. Create a function named **pattern** which takes an integer as an input and print the following pattern.

```
# Example 1
pattern(4)

# Output
1 2 4 8

# Example 2
pattern(7)

# Output
1 2 4 8 16 32 64 # Basically print upto 7 numbers
```

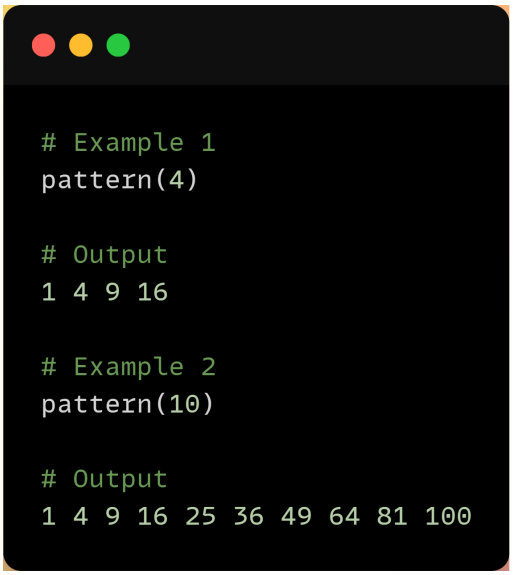
Q6. Don't create a function, just print the following pattern

1 11 111 1111 11111....n times **(Ask n from user)**

Q7. Keep asking numbers from user until the user enters **0**. Then display the average of all numbers.

```
Enter a number (enter 0 to finish): 54
Enter a number (enter 0 to finish): 14
Enter a number (enter 0 to finish): 63
Enter a number (enter 0 to finish): 98
Enter a number (enter 0 to finish): -111
Enter a number (enter 0 to finish): 521
Enter a number (enter 0 to finish): 3
Enter a number (enter 0 to finish): 0
The average of the entered numbers is: 91.71428571428571
```

Q8. Write a function named **pattern** which accepts an integer **n** as an argument. Then print the following pattern.



```
# Example 1
pattern(4)

# Output
1 4 9 16

# Example 2
pattern(10)

# Output
1 4 9 16 25 36 49 64 81 100
```

Q9. Ask a number from user. Print if that number is prime or not, use functions.

Q10. Ask a number from user **n1.** From **1 to n1,** print how many prime numbers are there.