UNIT 3

1.Write a python program to perform arithmetic calculator using functions.

```
def add(x, y):
    return x + y

def subtract(x, y):
    return x - y

def multiply(x, y):
    return x * y

def divide(x, y):
    if y != 0:
        return x / y
    else:
        return "Cannot divide by zero"

def calculator():
    print("Simple Arithmetic Calculator")
    print("Select operation:")
    print("1. Addition")
    print("2. Subtraction")
    print("3. Multiplication")
    print("4. Division")

    choice = input("Enter choice (1/2/3/4): ")

    num1 = float(input("Enter first number: "))
    num2 = float(input("Enter second number: "))

    if choice == '1':
        result = add(num1, num2)
        print(result)
    elif choice == '2':
        result = subtract(num1, num2)
        print(result)
    elif choice == '3':
        result = multiply(num1, num2)
        print(result)
    elif choice == '4':
        result = divide(num1, num2)
        print(result)
    else:
        print("Invalid input. Please enter a valid choice.")

if __name__ == "__main__":
    calculator()
```

OUTPUT:

Simple Arithmetic Calculator

Select operation:

1. Addition

2. Subtraction

3. Multiplication

4. Division

Enter choice (1/2/3/4): 1

Enter first number: 45

Enter second number: 23

68.0

2.i)Write a python program to generate a duck number up to 656 using functions[CO3,K2] ii) Write a program to check whether entered string is palindrome or not using function[CO3,K2]

```
def is_duck_number(n):
    n_str = str(n)
    return '0' in n_str and n_str.index('0') != 0


def generate_duck_numbers(limit):
    duck_numbers = []
    for num in range(1, limit + 1):
        if is_duck_number(num):
            duck_numbers.append(num)
    return duck_numbers


if __name__ == "__main__":
    limit = 656
    duck_numbers = generate_duck_numbers(limit)

    print(f"Duck numbers up to {limit}:")
    print(duck_numbers)
```

OUTPUT:

Duck numbers up to 656:

[10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 420, 430, 440, 450, 460, 470, 480, 490, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 520, 530, 540, 550, 560, 570, 580, 590, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 620, 630, 640, 650]

```
def palindrome(s):
    s = s.lower().replace(" ", "")
    return s == s[::-1]
if __name__ == "__main__":
    user_input = input("Enter a string: ")
    if palindrome(user_input):
        print("The entered string is a palindrome.")
    else:
        print("The entered string is not a palindrome.")
```

OUTPUT:

Enter a string: welfare

The entered string is not a palindrome.

3.i) Printing Numbers in a Simple Triangle Pattern

```
n=int(input("enter the no of rows:"))
for i in range(1, n + 1):
    for j in range(i):
        print(i, end=" ")
    print()
```

OUTPUT:

enter the no of rows:5

1

2 2

3 3 3

4 4 4 4

5 5 5 5 5

```
n = int(input("enter the no of rows:"))
for i in range(1, n + 1):
    for j in range(i, 0, -1):
        print(j, end=" ")
    print()
```

OUTPUT:

enter the no of rows:5

1

2 1

3 2 1

4 3 2 1

5 4 3 2 1

4. Write a program to print the following pattern

```
n = int(input("enter the no of rows:"))
for i in range(1, n + 1):
    print("*" * i)
```

OUTPUT:

enter the no of rows:5

*

**

***

****

*****

```
n = int(input("enter the no of rows:"))
for i in range(n, 0, -1):
    print("*" * i)
```

OUTPUT:

enter the no of rows:4

****

***

**

*

6. Write a program that generates the arithmetic progression up to n terms and store it in a list and find the sum of all the list elements present in the list? [CO3,K3] sample input : 10 2 3 SAMPLE OUTPUT 2 5 8 11 14 17 20 23 26 29 155 EXPLANATION First line of the input is n Second line of the input is starting value of the sequence Third line of the input is difference between any two terms in the sequence General form AP is a , a+d ,a+2d, a+3d…

```
def generate_ap(n, a, d):
    ap_list = [a + i * d for i in range(n)]
    return ap_list


if __name__ == "__main__":
    n = int(input("Enter the number of terms: "))
    a = int(input("Enter the first term: "))
    d = int(input("Enter the common difference: "))
    ap_sequence = generate_ap(n, a, d)
    print("Generated Arithmetic Progression:")
    print(" ".join(map(str, ap_sequence)))
    sum_of_elements = sum(ap_sequence)
    print("Sum of elements:", sum_of_elements)
```

OUTPUT:

Enter the number of terms: 10

Enter the first term: 2

Enter the common difference: 3

Generated Arithmetic Progression:

2 5 8 11 14 17 20 23 26 29

Sum of elements: 155