**PROJECT REPORT**

**ON**

**PYTHON LABOTORY**

****

**Submitted By :**

**Asidur Rahman (CS23BCAGN057)**

**BCA IV SEMISTER**

**SCHOOL OF COMPUTING SCIENCES**

**The Assam Kaziranga University , Jorhat, Assam**

**May 2025**

**Table of Contents**

1. Introduction
2. Program 1: Arithmetic and Quadratic Operations
3. Program 2: Linear Equation Solver
4. Program 3: Graphical Representations
5. Program 4: Function Implementation
6. Program 5: Snake Game using Tkinter

**1. Introduction**

This Python project covers a range of basic to intermediate programming tasks such as math operations, equation solving, visualization, and game development using Tkinter.

2. Arithmetic and Quadratic Operations

Code:

import cmath

def arithmetic\_operations(a, b):

print("Addition:", a + b)

print("Subtraction:", a - b)

print("Multiplication:", a \* b)

print("Division:", a / b if b != 0 else "Division by zero error")

def solve\_quadratic(a, b, c):

d = (b \*\* 2) - (4 \* a \* c)

root1 = (-b - cmath.sqrt(d)) / (2 \* a)

root2 = (-b + cmath.sqrt(d)) / (2 \* a)

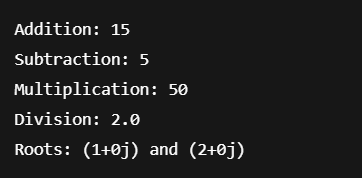
print(f"Roots: {root1} and {root2}")

# Example usage

arithmetic\_operations(10, 5)

solve\_quadratic(1, -3, 2)

**Output:**



3. Linear Equation Solver

Code:

def solve\_linear(a, b):

if a == 0:

print("No solution" if b != 0 else "Infinite solutions")

else:

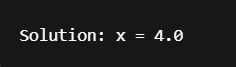
x = -b / a

print(f"Solution: x = {x}")

# Example

solve\_linear(2, -8)

Output :



4. Graphical Representations

Code :

import matplotlib.pyplot as plt

# Define coordinates for star

x = [0, 2, 4, 1, 3, 0]

y = [0, 4, 0, 2.5, 2.5, 0]

# Plotting the star

plt.figure(figsize=(5,5))

plt.plot(x, y, marker='o', color='gold')

plt.fill(x, y, 'gold', alpha=0.7)

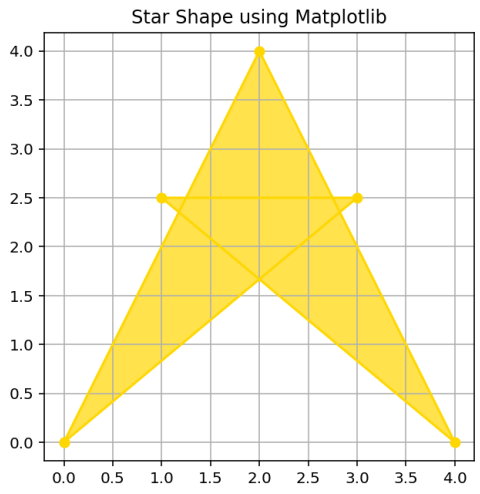
plt.title("Star Shape using Matplotlib")

plt.axis('equal')

plt.grid(True)

plt.savefig("/mnt/data/star\_graph.png")

plt.show()

Output :

5. Function Implementation (Factorial)

Code :

def factorial(n):

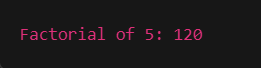
if n == 0 or n == 1:

return 1

return n \* factorial(n - 1)

print("Factorial of 5:", factorial(5))

Output :



1. Formatted App: Snake Game using Tkinter

Code :

import tkinter as tk, random as r

W, H, S, DIR = 500, 500, 20, "right"

class Game:

def \_\_init\_\_(self, root):

self.c = tk.Canvas(root, W, H, bg="black")

self.c.pack()

self.snake = [[0,0],[S,0],[2\*S,0]]

self.snake\_squares = [self.c.create\_rectangle(x,y,x+S,y+S,fill="#0f0") for x,y in self.snake]

self.food = None

self.new\_food()

root.bind("<Key>", self.dir\_change)

self.move()

def new\_food(self):

while True:

self.f = [r.randrange(0,W,S), r.randrange(0,H,S)]

if self.f not in self.snake:

if self.food: self.c.delete(self.food)

self.food = self.c.create\_oval(\*self.f, self.f[0]+S, self.f[1]+S, fill="#f00")

break

def move(self):

global DIR

x,y = self.snake[0]

if DIR=="left": x-=S

if DIR=="right": x+=S

if DIR=="up": y-=S

if DIR=="down": y+=S

if [x,y] in self.snake or not (0<=x<W and 0<=y<H):

self.c.delete(tk.ALL)

self.c.create\_text(W//2,H//2,text="GAME OVER",fill="red",font=("Arial",30))

return

self.snake.insert(0,[x,y])

self.snake\_squares.insert(0,self.c.create\_rectangle(x,y,x+S,y+S,fill="#0f0"))

if [x,y] == self.f: self.new\_food()

else:

tail = self.snake.pop()

self.c.delete(self.snake\_squares.pop())

self.c.after(100,self.move)

def dir\_change(self,e):

global DIR

opposites = {"Left":"Right","Right":"Left","Up":"Down","Down":"Up"}

if e.keysym in opposites and opposites[e.keysym] != DIR.capitalize():

DIR = e.keysym.lower()

tk.Tk().title("Snake Game")

Game(tk.Tk()).c.master.mainloop()

Output :

