**LAB#03 – ALL TASKS COMBINED**

**SERVER.PY**

import socket as sock

import threading

grading\_schema = [

    (90, "A+", 4.00),

    (86, "A", 4.00),

    (82, "A-", 3.67),

    (78, "B+", 3.33),

    (74, "B", 3.00),

    (70, "B-", 2.67),

    (66, "C+", 2.33),

    (62, "C", 2.00),

    (58, "C-", 1.67),

    (54, "D+", 1.33),

    (50, "D", 1.00),

    (0, "F", 0.00),

]

def calculate\_grade(marks):

    for threshold, grade, gpa in grading\_schema:

        if marks >= threshold:

            return grade, gpa

    return "F", 0.0

def handle\_client(conn, addr):

    conn.sendall(b"Welcome to FAST-NUCES Karachi Campus CGPA Calculator!\n")

    student\_id = conn.recv(1024).decode().strip()

    conn.sendall(b"Enter number of subjects: ")

    num\_subjects = int(conn.recv(1024).decode().strip())

    total\_points, total\_credits = 0, 0

    results = []

    for i in range(num\_subjects):

        conn.sendall(f"Subject {i+1} - Enter credit hours: ".encode())

        ch = int(conn.recv(1024).decode().strip())

        conn.sendall(f"Subject {i+1} - Enter marks (out of 100): ".encode())

        marks = int(conn.recv(1024).decode().strip())

        grade, gpa = calculate\_grade(marks)

        total\_points += gpa \* ch

        total\_credits += ch

        results.append((ch, marks, grade, gpa))

    cgpa = total\_points / total\_credits if total\_credits > 0 else 0.0

    response = f"\nStudent ID: {student\_id}\n"

    for idx, (ch, marks, grade, gpa) in enumerate(results, 1):

        response += f"Subject {idx}: Credit Hours={ch}, Marks={marks}, Grade={grade}, GPA={gpa}\n"

    response += f"Overall CGPA: {cgpa:.2f}\n"

    conn.sendall(response.encode())

    with open("cgpa\_log.txt", "a") as f:

        f.write(response + "\n")

    conn.close()

def start\_server():

    server = sock.socket(sock.AF\_INET, sock.SOCK\_STREAM)

    server.bind(("127.0.0.1", 5555))

    server.listen(5)

    print("Server running on 127.0.0.1:5555")

    while True:

        conn, addr = server.accept()

        threading.Thread(target=handle\_client, args=(conn, addr)).start()

if \_\_name\_\_ == "\_\_main\_\_":

    start\_server()

**CLIENT.PY**

import socket as sock

def run\_client():

    client = sock.socket(sock.AF\_INET, sock.SOCK\_STREAM)

    client.connect(("127.0.0.1", 5555))

    print(client.recv(1024).decode(), end="")

    student\_id = input("Enter your Student ID: ")

    client.sendall(student\_id.encode())

    print(client.recv(1024).decode(), end="")

    num\_subjects = input()

    client.sendall(num\_subjects.encode())

    for \_ in range(int(num\_subjects)):

        print(client.recv(1024).decode(), end="")

        ch = input()

        client.sendall(ch.encode())

        print(client.recv(1024).decode(), end="")

        marks = input()

        client.sendall(marks.encode())

    print("\n--- Reslt ---")

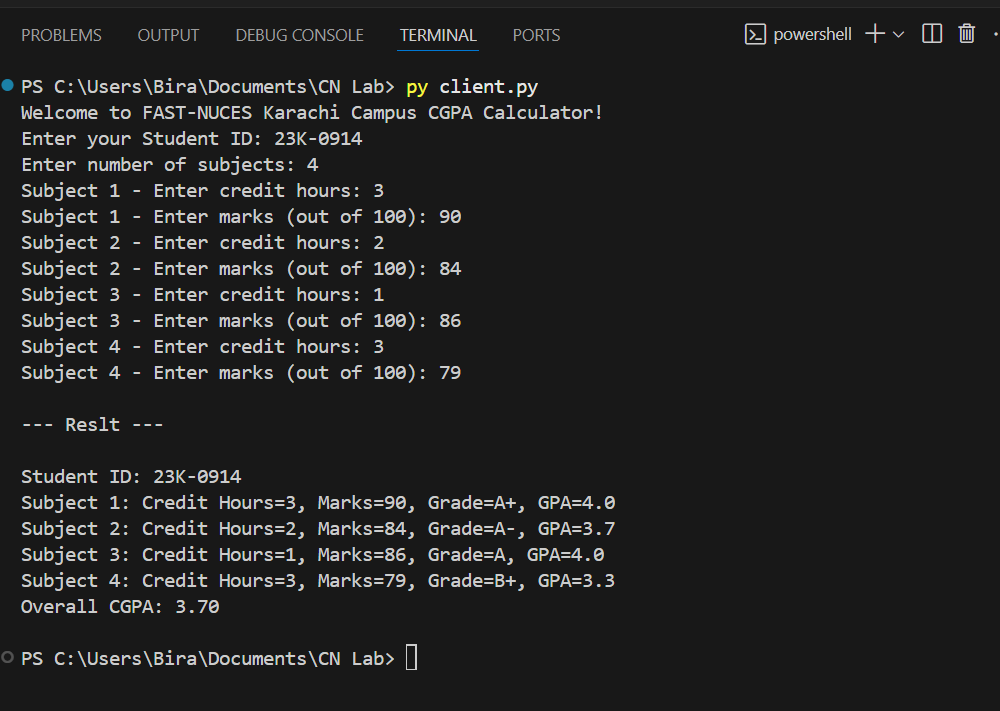
    print(client.recv(4096).decode())

    client.close()

if \_\_name\_\_ == "\_\_main\_\_":

    run\_client()

**OUTPUT**

****