Assignment 1

Total: 100 Marks

Task 1: Understanding Parallelism (10 Marks)

Write a short explanation (200-300 words) on:

a) The difference between **multiprocessing** and **multithreading**.

b) When to use **process-based parallelism** vs. **thread-based parallelism** in Python.

Task 2: Implementing a CPU-Bound Task Using Multiprocessing (30 Marks)

Write a Python program that computes the sum of squares for a large range of numbers.

a) Implement **both sequential and parallel versions** using *multiprocessing*.

b) Measure the execution time for both versions using *time.time()*.

c) Compare the performance and write a short conclusion.

Hints:

- Use *Pool* from *multiprocessing* to create worker processes.
- Try calculating the sum of squares for *range(1, 10**7)*.

Task 3: Implementing an I/O-Bound Task Using Multithreading (30 Marks)

Simulate an **I/O-bound** task by downloading multiple web pages using Python's threading module.

a) Implement a function that fetches a webpage (e.g., https://example.com).

b) Run it **sequentially** and then using *ThreadPoolExecutor*.

c) Measure the execution time for both approaches and compare performance.

Hints:

- Use *requests.get(url)* for fetching web pages.

- Use *concurrent.futures.ThreadPoolExecutor* for threading.
- Fetch 10 different URLs and compare execution times.

Task 4: Combining Multiprocessing and Multithreading (30 Marks)

Extend Task 3 by **combining multiprocessing and multithreading**:

a) Use *multiprocessing* to split a list of 100 URLs across **4 processes**.

b) Each process will use *threading* to fetch **10 URLs in parallel**.

c) Measure the total execution time and compare it to Task 3.

Hints:

- Use *multiprocessing.Pool* to distribute the workload.
- Use *ThreadPoolExecutor* inside each process.

Submission Guidelines:

i. Submit a **Python script (.py file)** for each task.

ii. Include a short report (PDF) summarizing results and performance comparisons.

iii. Provide screenshots of execution times.

Bonus: Describe when to use Multiprocessing and when to use Multithreading based on your understanding in light of Task3-4. (**10 Marks**)