|  |  |  |
| --- | --- | --- |
| **جامعـة العلـوم التطبيقية الخاصة**  **كلية تكنولوجيا المعلومات**  **مشروع (#2) 2024-2**    **Student Name:**  **Student ID:**  **□ CS □ SE □ SEC □ DSAI** |  | **APPLIED SCIENCE UNIVERSITY (ASU)**  **FACULTY OF INFORMATION TECHNOLOGY (FIT)**  **Project (#2) 2024-2**  **Subject:** *Parallel programming 1301421*  **Instructor:** |

**Please note that this Project and with a total of 15 points.**

**Question 1: [15 points]**

Your task is to build a **Java-based encryption tool** that reads data from a **text file**, encrypts it using a simple symmetric algorithm (e.g., Caesar Cipher, XOR, or AES), and writes the encrypted content to an output file. You will implement both **sequential** and **multithreaded** versions, measure performance, and simulate **message passing** between threads.

### **Part A – Sequential Implementation [4 Points]**

* Implement encryption in **(sequential)** Java program.
* Choose a basic encryption algorithm (e.g., Caesar Shift, XOR, or AES).
* Encrypt the content and save it .
* Measure and print the **execution time in milliseconds**.

### **Part B – Parallel Implementation (5 Marks)**

* Convert your program to run in parallel
* Implement and test the program using:
  + 4 threads
  + 8 threads
  + 16 threads
  + 32 threads
* Each thread must handle a subset of the dataset.
* Measure and print the execution time for each version.

### **Part C – Performance Comparison & Plot (2 Marks)**

* Compare execution times for all versions (sequential, 4, 8, 16, and 32 threads) and generate a **bar chart** showing the execution time for each version.

### **Part D – Message Passing Simulation (4 Marks)**

* Simulate thread communication using **message passing**:
  + Each thread finds the best result in its range and **passes** the result to the main thread.
  + Use BlockingQueue or any other synchronized structure for communication.
  + The main thread collects all results and chooses the final best recommendation