Computer Science Division
Department of Mathematics
Faculty of Science

Level 4- Semester 1
Course: COMP 403
Date: 2025

Q1- Write a program to count the odd and even numbers from a given array by using multi-threading.

Q2- Ask the user to enter the array size, each index filled with a number. Count the odd and even digits in the number and save them in another two arrays (EvenArr and OddArr). The last index in EvenArr will be the summation of even digits in all numbers. Similarly, OddArr.

Array	2345	62845	572	4444	36739	
EvenArr	2	4	1	4	1	50
OddArr	2	1	2	0	4	47

Q3- Write a program to find the maximum element from a given array using multi-threading.

Q4- Ask the user to enter two numbers and give the GCD using multi-threading.

For example:

1. Let x and y be natural numbers stored in memory cells M_1 and M_2, respectively, of the RAM model. The greatest common divisor of numbers x and y, GCD(x, y), can be computed by the Euclidean algorithm:[31]

```
while x ≠ y do
    while x > y do x := x − y end while;
    while y > x do y := y − x end while;
end while;
```

Give the **RAM programs** on the low and middle level of abstraction (see Figure 2.2a and 2.3a) to compute the GCD(x, y). The desired result should appear in cell M_1.

Q5- Ask the user to enter the matrix size and fill the matrix randomly. Search for duplicate numbers by using threads and replace the duplicate numbers with the thread number being searched with. Compute its running time, speed up, cost, and its efficiency.

Rules:

- **ParallelTime** = end_time - start_time
- **SerialTime** = T1_end_time - T1_start_time
- // Number of threads used
- **Speedup** = SerialTime / ParallelTime
- **Cost** = NumThread * ParallelTime
- **Efficiency** = Speedup / NumThread