Kingdom of Saudi Arabia
Al-Imam Mohammad Ibn Saud Islamic University
Computer and Information Sciences College
Computer Sciences Department

CS222 – Operating Systems
Programming Assignment - 1
Instructors: Ghada AlOsaimi, Haifa AlDhayel

Objectives:

The aim of this assignment to help the students to understand:
(1) Creation, termination, and controlling of child processes in Linux environment.
(2) How to write multithreaded programs and practice the concepts learned in class.
(3) Implementing multithreaded programs using Pthreads libraries on Linux environment.

Instructions:

1) Download the programming assignment 1 sheet from Blackboard.
2) Solve all included problems.
3) Submit your answers as a zipped folder contain **source code files of both problems** in Blackboard under Assignment Submission of Assignment 1 at Programming Assignments tab.
4) Due date of assignment submission is no later than **20th February 2021 (11:00 pm)**.
5) This assignment is a group work; no more than two members in a group.
6) Note that all group members should submit the work through their Blackboard account.
7) You are not allowed to submit the work several times, submit the work once you are committed.

Problem 1:

A Latin square is a square grid of size n x n, with a number written in each cell, such that each number between 1 and n appears exactly once in each row and once in each column. Latin squares come in many sizes; this assignment consists of designing a multithreaded application that determines whether the solution to a 5 x 5 Latin square is valid. Figure 1 presents an example of a valid Latin square.

1	2	3	4	5
2	1	4	5	3
4	3	5	1	2
3	5	1	2	4
5	4	2	3	1

Figure 1: Valid Latin Square

There are several different ways of multithreading this application indicated below you are free to choose the way you prefer. You can add more threads as you are needed.

- A thread or five threads to check that each column contains uniquely the digits 1 through 5.
- A thread or five threads to check that each row contains uniquely the digits 1 through 5.

The parent thread (main) is responsible to (1) create other threads and (2) print the result which is validity of square. Declare a 5 x 5 square that you would check its validity as a global variable with valid values. Do not forget to support your code with proper comments (i.e., **documentation**) and use all functions that are essential to complete the application (create, join and exit threads) properly.

Problem 2:

Use the same source code of previous problem and create a child process using fork() system call to check its version of the global 5 x 5 square after changing its values to invalid within the child process. Make sure the child process must run and finish its execution before parent process. Show validity of the square in both processes parent (valid) and child (invalid) along with their IDs.

Do not forget to support your code with proper comments (i.e., **documentation**) and use all required system calls (fork, wait, exit) functions that are essential to complete the application properly.