UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Department of Computer Science
COS 226 - Concurrent Systems

Practical 5

- **Date issued:** 30 September 2022
- **Deadline:** 06 October 2022, 8:00 PM
- This practical consists of 2 task. Read each task **carefully!**

1 Introduction

1.1 Objectives and Outcomes

This practical aims to further explore synchronization by evaluating the role of locking via Linked-lists.

You must complete this assignment individually. Copying will not be tolerated.

1.2 Submission and Demo Bookings

You are NOT provided with any skeleton code for this practical, you will have to implement everything yourself.

Submit your code to **clickup** before the deadline.

You will have to demonstrate each task of this practical during the **physical** practical lab session. So be sure to create copies of your source code for each task separately. Booking slots will be made available for the practical demo.

1.3 Mark Allocation

For each task in this practical, in order to achieve any marks, the following must hold:

- Your code must produce console output. (As this is not marked by fitchfork, formatting is not that strict)

- Your code must not contain any errors. (No exceptions must be thrown)
- Your code may not use any external libraries apart from those highlighted in the textbook.
- You must be able to explain your code to a tutor and answer any questions asked.

The mark allocation is as follows:

Task Number	**Marks**
Tasks combined	10
Total	10

2 Practical Requirements

You are required to simulate a security protocol for access control. This time the protocol is for access to an art gallery.

The access protocol works as follow:

- The gallery has 5 entrances/access points each controlled by a security personnel.
- When a person enters the gallery they are given a random time between 100 and 1000 ms which they can be in the gallery. When the time lapse they will be escorted by security out of the gallery.
- People (art lovers) will have to form a queue at each access point and the security personnel will only allow one person into the gallery after every 200 ms, it can be later but not earlier
- Each person will have to leave the gallery using the access point/door they used when they came in.
- Since there is only one security guard at each access point, the guards will be excused if times lapse while they are busy with other duties

2.1 Tasks

Implement the following tasks which simulates the above protocol

1. Task 1: Coarse-Grained Synchronization
2. Task 2: Fine-Grained Synchronization
3. Task 3: Optimistic Synchronization

Note:

- A thread will simulate a security personnel responsible for a specific access point/entrance and the gallery will be the critical section.
- At each access point/entrance there are 10 people
- **NB! Please note that you must be able to explain the different implementations in your demo as most of the code is already provided.**

2.2 Output

The following output is expected:

- Every-time a person enters the gallery, print out the person-name and the time they are allocated
  - [Thread-name]: added ([Person-number],[time-left])
  - Example: Thread-1: added (P-1, 12ms)
- Everytime a person leaves the gallery, print out the linked list
  - [Thread-name]:([Person-number],[time-left]),([Person-number],[time-left]), ...
  - Example: Thread-1: (P-1, 0ms), (P-4, 75ms)