Diana Kocsis

Professor Sherif Khattab

CS1550: Introduction to Operating Systems

6 July 2020

Project 2 Solution

My solution to Project 2 is fair because the processes waiting on a condition are placed inside a condition variable queue. Whenever another process signals, it looks into the condition variable queue and resumes the next process that was suspended. This means that all processes will execute and finish at some point. Since the visitor processes can only tour the museum for 2 seconds, the visitors will eventually leave, the tour guides will be able to leave, new tour guides will be able to enter, and new visitors will be able to enter.

My solution is also deadlock free because no 2 processes are waiting for each other's resources. As an example, I make sure to wake up any tour guides waiting outside the museum before the visitor begins to wait. If the visitor process didn't signal/wake up the tour guides, the tour guides would be waiting for a visitor and the visitor would be waiting for a tour guide, resulting in a deadlock situation.

Lastly, my solution is starvation free because there are no processes that can wait indefinitely. As an example, if a tour guide opens the museum, he will signal all of the processes that were waiting for a tour guide in the condition variable queue. If he awakens more than 10 (the maximum amount per tour guide) the rest will have to wait again and go back into the queue until another tour guide arrives. The next tour guide that opens the museum will then resume the next visitor process that was suspended in the queue. No processes will ever wait indefinitely.