COMP 2004

Assignment 5

1.

a. What data have a race condition?

The variable TOP can have a race condition. Because it is shared among the three functions.

b. Describe how could the race condition be fixed.

To fix this race condition we need proper locks/atomic operations to be used to ensure that only one thread can access and modify the shared variables at a time. Using a lock to enforce mutual exclusion. Before accessing the top variable, a thread would acquire the lock, perform the necessary operation, and then release the lock. This would ensure that only one thread can modify the top variable at a time, preventing race condition.

2. Throughput in the readers-writers problem is increased by favoring multiple readers as opposed to allowing a single writer to exclusively access the shared values. On the other hand, favoring readers could result in starvation for writers. A method to solve it can be by keeping timestamps associated with waiting processes. When a writer is finished with its task, it would wakeup the process that has been waiting for the longest duration.

3.

a. (15%) Identify the data involved in the race condition.

The data involved in the race condition is the available_resources variable.

b. (15%) Identify the location (or locations) in the code where the race condition occurs.

The race condition occurs in the decrease_count() function. Because when multiple threads execute this function concurrently and may access and modify the shared variable available_resources in an mixed order, leading to leading to inconsistent results and race condition.