Assignment 3: Deadlock Avoidance

Austin Johns

CST-315

William Hurst

30 March 2020

**Definition of Deadlock Avoidance**

In an operating system, deadlock is when there are two process blocking each other because both processes are holding a resource and waiting for another resource from another process. A deadlock between processes can be dangerous in an operating system because it can cause a total freeze of processes and cause a process to fail. The best way to mitigate this problem is with deadlock avoidance.

One of the best ways to go about deadlock avoidance is by eliminating hold and wait. This can be done by loading all of the resources necessary to the process before the process is executed. This eliminates the hold and wait because no resources are fetched. Another way to go about deadlock avoidance is by ending circular wait. With a number assigned to each resource, a process can request a resource and the number will increment/decrement based on its usage.

For this project, the logical process is represented in the flowchart below. The best way to go about this is by tracking time and incrementing a number representative for each process and resource. This allows the process to only use the resources that are available. The resources are only available once a process has released them, incrementing the number.

**A close up of text on a white background

Description automatically generatedLogic in Code**

**Program Execution Results**

**A screenshot of a cell phone

Description automatically generated**