Name: Garrett Bearss

Documentation

The Banker's Algorithm works but allocating each process a certain number of resources. It then uses the max resources allocated to each resource to determine a pattern or sequence in which each process can run without deadlock occurring. I first started off in my code by allocating each of the different resources based on the information given from the input file using the function getData(). After that was complete, I then calculate the needed resources of each process by taking the max resource of a process and subtracting it by the allotted. I then used an array to keep track of which processes still need to be finished and start looking for the safe sequence. I use a loop to find which can run on the current set of resources. I do this by checking if the current resources plus the allocated is enough to satisfy the needed resources of a process. After that I mark the process as finished and continue looping until each process has been sorted through to find a safe sequence. The loop only continues for the number of processes and if not, every process has had their needs satisfied then no safe sequence can be found. Depending on the result the program may print the safe sequence if there is one or may print one could not be found.

The following commands are used to run the program:

$ g++ main.cpp -o main

$ ./main