Western Oregon University

Memo

|  |  |
| --- | --- |
| To: | Dr. Cordova |
| From: | Justin Davis |
| Date: | 1/30/2022 |
| Re: | Project Proposal |

The algorithm I have been looking into and would like to work on implementing is the Priority Based Scheduling algorithm, and I am planning to use C# since that is what we have done our demos in and I am most comfortable with. By definition, this algorithm determines which task needs to be done the most urgently based on its assigned priority ranking, and runs the highest first. Those of lower priorities are pushed to the back of the line. If there are multiple tasks that need to be ran that have the same level of priority, then the Round-Robin algorithm is used to complete those. The priority of a task can be determined by considering things like the memory or time requirements but isn’t limited to those two factors. I feel like calculating the amount of memory used would potentially be the easiest to determine, and if the algorithm happened to be used on a system constrained by its memory this would be a very important thing to consider. So, I would want to approach it with a memory-based, Round-Robin, Priority Based Scheduling algorithm.

The most vital input, other than the program to be ran, from what I can tell would be the amount of memory a program is going to require. I feel like this is something that would have to be provided beforehand or somewhere at the beginning of the program, or somehow ascertained by a program looking at the lines of code first and determining what lines are going to populate memory; which I also think would be rather time consuming, unneeded and hard to make in this scope. The other inputs I could see being needed is the output location of the program that’s been provided. Just as we did with the fork demo, the name of the program, and the location of its output is necessary. This would then create the single output of the program, which would be some kind of binary file that could then be run to see the programs results.