CS475: Lab 7 Code Review 😀

# Questions to answer:

Spencer Gariano, Trenten Nogle, Finn McClone, Jonathan Olderr

**0.      Who did you work with? But why not**

Jeremy, David, and Samira

**1.      How is time defined?**

The system defined the time. (Future: Each processor has a given amount of time that it can run, and the processor…?)

**2.      What kinds of things can happen at each “moment”?**

Process information such as arrival time, ID etc is created. Processes are updated, and the processes are switched between the ready queue, and wait queue. At some moments it has to find which process to run.

**3.      What happens when the parameters of the simulation changes? Do the results make sense?**

Could not grab the results to find the different simulation data, but assumes that the parameters would cause the simulation outcome to change. The results do make sense.

**4.      Were there any bugs or issues that you fixed?**

Wouldn’t recognize cout or vectors (std library objects) in a certain vscode, when on a different computer, worked fine.

Program would loop forever because the ready queue wasn’t being shortened or decremented.

**5.      Explain how each data structure is used.**

deque(s) to represent the ready and wait queue.

Struct for PCBs and processes.

Vectors of object pointers

Maps to map PID to process

**6.      What alternative data structures were considered? Why were they not chosen?**

Group thought about vectors or queues instead of the deques for ready and wait data structure. Deques were decided because the group thought that they would fit more data than a vector. And deque would handle front and back ends of the data structures.

**7.      What alternative data structures would you have also considered?**

Group made the consensus that they chose the correct data structures, other than possibly lists or stack. Some possible data structures to implement time.

**8.      Is memory managed correctly (e.g. are pointers correctly deallocated)?**

Whenever a process / PCB is created dynamically, the CPU has a deconstructor which deletes the process and PCB when needed.

**9.      Are exceptions handled correctly?**

Did not include any try/catches to handle errors.

**10. Other questions?**

In the future they talked about the next steps being implementing a time quantum. Implementing file I/O. Adding the ability to scale larger processes and analyze those results, and having more processes.