CS 411 Operating Systems II Project 1

Implement the RR and FIFO process scehduling algorithm

Li Li

1 What do you think the main point of this assignment is?

As the first assginment of the class, I think it would be a more or less intro assginment. There are many peraration need to be done before go further in this class. These may include:

- *SVN* Mostly because many people are not familiar with it. Since most of the works done in the course are group works, SVN is vitial to make sence in group projects as well as future works.
- *Squidly* It gives us a sence to apply what we learned from textbook in real world. It is much accomplishments to figure out how to modify a OS.
- *First theme: process scheduling* Including how it works in real world OS manners and how to make a modification.

2 How did you personally approach the problem? Design decisions, algorithm, etc.

When I first go to think over the problem, I thought it may need to write clumsy codes to represent FIFO and RR algorithms. But then it became a big problem to went through in detail. And when we went to the TA's, he implied there are some lines erased from the original one. So we went to offical kernel.org to download the source codes and diffed the codes to figure out the differences between them. The rest is to figure out how it does the job.

3 How did you ensure your solution was correct? Testing details, for instance.

Another issue is how we make sure the FIFO and RR mechanisms work fine in the modified kernel. My initial idea is to create a branch of processes and print their PIDs when running. I implement the algorithm in threads, because I heard the threads are kind of the same as process in linux. And I want to write a program to make sure it is true. After testing, the result of RR shows like printing one thread multi-times and then running to another and then another till the end. It is weird. But after discussing with my fellow groupmate Nick, I figure out the kernel is actually working correctly. The problem was the quantum of the thread is too long. And in the quantum, the thread ran over several times and returned. So the solution would be either minimizing the quantum or giving the thread some more sophisticated works to do. My work was to give a math problem: finding fibonacci number. And finally I got the expected results.

4 What did you learn?

As I mentioned in beginning, I learned how to work on SVN, how to make a modified kernel and made sure that threads sceduling worked similar to processes in linux kernel. In addition, group debugging is, I think, much more easy than individual debugging.