CSC3002F 2023

Operating Systems Part 2

Lecturer: Michelle Kuttel

Assignment: **Process Scheduling**

May 2023

Introduction

This assignment aims to increase your understanding of **CPU process scheduling,** particularly the different **metrics** by which schedulers are compared, and to give you some experience with testing scheduling algorithms using a **simulator**.

You are given a Java software **framework** for simulating process scheduling and **two working kernels** for scheduling: a **First Come First Served** (FCFS) kernel and a **Round Robin** (RR) kernel. Using this framework, you will profile both **FCFS** and **RR scheduling** across a range of metrics. In this assignment we will focus on **CPU usage**, **waiting time** and **turnaround time**. You will investigate the effect of **context switching time** and **system call time** on these metrics. For RR, you will also experiment with **different time quanta**. The size of the **time quantum** q has an effect on RR scheduling: it must neither be too large, nor too small. We want to ensure that:

a. the time quantum is **large enough** relative to the time for a context switch, to ensure good **CPU utilisation**; and
b. the time quantum is **small enough** to ensure both **good turnaround time** and waiting time.

If the time quantum is too large, RR scheduling is expected to **degenerate to an FCFS policy**.

Assignment Specifications

Your task is to use the **simulator** and a **particular job workload** to compare the performance of the **`RRKernel`** and **FCFSKernel** schedulers for a given workload of 18 processes. For the **`RRKernel`** you must identify the **optimal time quantum**, by generating **graphs** time of quantum size versus:

- **CPU utilisation;**
- **average turnaround time** for the processes that ran and
- **average waiting time** for the processes that ran**.**

The **points for the FCFS algorithm** should also be plotted **on the graphs**.

You must experiment with different system call times and context switching times. A **system call time of 5** time units and a **context switch time of 20** time units is a good place to start, but can you should experiment with at least 2 other settings.

It will help if you write **shell scripts** to automate the many runs you will need and Python scripts to extract the data from the tracing file output by the simulator.

Then must submit a **short report** where you **explain** what you see in these graphs and **draw conclusions** – where you see best CPU utilization, where is the best turnaround time and when RR becomes FCFS (note that just the graphs is not enough, you must explain). You should also explain the effects of different context switching times and system call times.

The Simulation Framework

Computer simulation is used to model the behaviour of a complex system. The intention is that the simplified model used to create the simulation has predictive properties i.e. it tells us something about a real world system. In this assignment, we model the behaviour of an operating system kernel from the perspective of process scheduling. Because we're not interested in memory allocation, file handing, and device drivers and so on, these do not appear in the model.

The framework may be downloaded from the assignment page. The '`.zip`' file contains a Framework directory/folder, which contains the Simulator Framework. This has an '`src`' directory, a directory called `processgenerator`, a directory/folder called '`tests`', a file called '`makefile`', a framework overview document called '`Framework.pdf`', and a document on creating simulation processes called '`ProcessGenerator.pdf`'. It also has a folder '`workload`' which contains a specified **simulator workload** `workloadConfig.cfg` that has 18 processes which you will use for this assignment (examine the file). The folder '`output`' is a good place to store the output of your runs.

The `src` directory contains a main simulation program called '**`Simulate`**', the working kernels, named '**`FCFSKernel`**' and '**`RRKernel`**', and the framework classes themselves in a folder called '**`simulator`**'.

To install the framework and run some first simulations, follow the instructions below.

Unzip the archive AssignmentMaterials2023.zip

In a terminal in the AssignmentMaterials2023 folder, do the following:

```
cd Framework
make all
```

- *This will build the framework, the simulator and the program generator (which you don't need for this assignment, but can use)*

```
java -ea -cp bin Simulate workload/run_fcfs_example.inp
```

- *This will run an example simulation with the FCFS schedule with, a system call cost of 1, and a context switch cost of 10.*

```
java -ea -cp bin Simulate workload/run_rr_q50_10_1_example.inp
```

- *This will run an example simulation with the RR scheduler with a system call cost of 1, a context switch cost of 10 and a quantum of 10.*

Now have a look at the framework overview document, '`Framework.pdf`'. This document contains an example of running a simulation using **`Simulate`**, the **`RRKernel`**, and a test workload from the **`tests`** directory.

'`ProcessGenerator.pdf`' is a document on creating simulation processes. You don't need this for this assignment, but can have a look.

See the framework overview document for a description of the profiling output produced from the simulator.

Submission

- A `pdf` document containing your report. It should be about 2-5 pages in length

NOTE: Because this is the end of term and we need to calculated DP, **no late handins** are accepted for this assignment: **you must submit by the specified deadline**.