COMP354 – Operating Systems
Project

Points: 100 possible (10% of your final grade)
Assigned Date: *Week 4*
Due Date: *Week 13 (Study Week)*

# 1 Assignment Objectives

- Gain practical experience with operating system concepts

**Submit either Part A or Part B**

# 2 Part A

For this project you will run a simulator for process scheduling. Your task will be to:

1. set up an experiment
2. run the experiment
3. analyze the results in a report.

If you chose to do part A, you must run **TWO** different experiments.

## *2.1 The simulator*

You can download the *Process (CPU) Scheduling* simulator from the course Moodle page that you will use for this project. You should first study the user manual, which is available on the website: *ps_doc.html*.

Steps to run the simulator:

1. You must have Java installed. In case you don't have it you can download Java from: http://www.java.com
2. You unzip *ps.zip* file into a folder (maybe into: comp354\project\)
3. You start the simulator from the command line using the *runps.bat* batch file (Windows) or *runps* script (Linux or macOS). As you can see in Figure 1, I have downloaded the simulator to the following folder: c:\Users\harald\comp354\project2

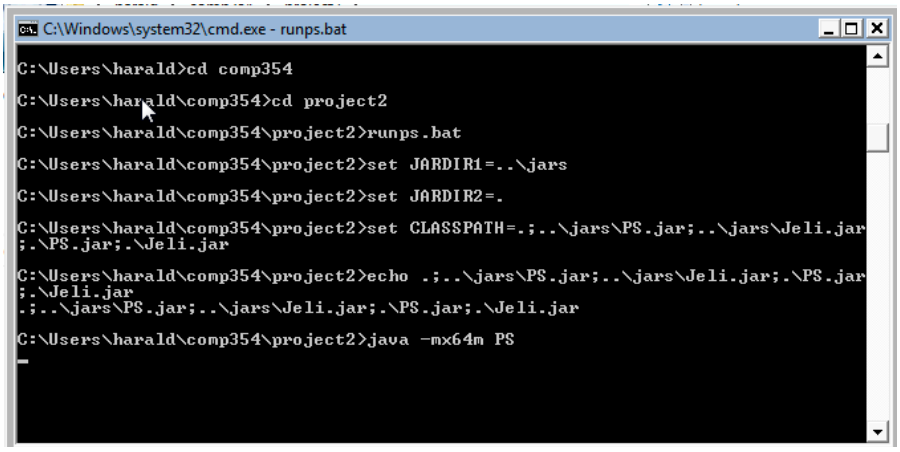

Figure 1: Running the simulator

## 2.2 Setting up the experiment

You must read the manual of how you will design your experiment. Each experiment consists of two files that you must create (you can also just modify the example files that come with the simulator). i.e. for your project you will have four files, because you must run two experiments.

### 2.2.1 myrun.run

In this file you specify the different types of processes that will be scheduled by the simulator. For the first experiment you will have two types of processes. For the second experiment you need to define three types of processes.

### 2.2.2 myexp.exp

In this file you will specify the experiment. For each experiment you will use four algorithms, namely: FCFS. SJF, PSJF, and RR. That means that in your myexp1.exp you will have six lines like:

```
name myexp1
comment This is the first experiment
run myrun1 algorithm FCFS key "FCFS"
run myrun1 algorithm SJF key "SJF"
run myrun1 algorithm PSJF key "PSJF"
run myrun1 algorithm RR 2.0 key "RR"
```

Note: RR takes one number as an argument. This number is the time-slice.

## 2.3 Running the experiment

In order to run an experiment you must first place the two files (myxxx.run and myxxx.exp) into the directory where you start the simulator in. You will then run all the runs for that specific experiment and take screen-shots of the Gantt-charts.

## 2.4 Requirements for the report

For each of the two experiments you must describe the following.

- Describe the configuration files for the specific experiment
- Analyze and describe the Running, Ready, and Waiting time for each run of the experiment
- Provide screen-shot of the Gantt-chart of each run
- Compare the Running, Ready, and Waiting time for the different algorithms in the different runs

In addition you must have the following:

- Table of contents
- An introduction where you describe how you used the simulator and how you designed your experiment.
- A conclusion where you describe what you learned from doing the experiment and any other information that you would like to add.
- You should also add the four files that you created for the two experiments as attachments to your report.

## 2.5 Grading Criteria

Your assignment will be evaluated after you submit a complete report (including table of contents, introduction, conclusion, etc). The following are the grading criteria:

| | |
|---|---|
| Report | |
| Table of contents, introduction, and conclusion | (5 points) |
| Designing the Experiment | |
| Setting up the required process groups (*.run file) | (10 points) |
| Setting up all the runs (*.exp file) | (10 points) |
| Running the Experiment | |
| Run all the runs for the two experiments | (10 points) |
| Providing the Gantt charts from the runs | (10 points) |
| Analyze the results | |
| Analyze and describe the Running, Ready, and Waiting time for each run of the experiments | (25 points) |
| Compare the Running, Ready, and Waiting time for the different algorithms in the different runs | (25 points) |
| Attachments | |
| Provide the experiment configuration files | (5 points) |
| Any other files you want to add | |

# 3 Part B

There are a number of different Programming Projects that you can choose from. You only need to do one of them.

## *3.1 Programming project 1*

For this project you will experiment with Kernel Modules on the Linux Kernel and system calls related to processes on a UNIX OS.

### 3.1.1 The Project

**Part A:** You have to complete the Programming Project at the end of Chapter 2 in the Operating Systems Concepts 10th Edition book.
**Part B: Pick 1 of the 4** Programming Project at the end of Chapter 3 in the Operating Systems Concepts 10th Edition book.

## *3.2 Programming project 2*

For this project you will develop an application that illustrates the different scheduling algorithms. Use either C or Java, on any platform (Linux, macOS, Windows, etc.)

### 3.2.1 The Project (Scheduling Algorithms)

You have to complete the Programming Project at the end of Chapter 5 in the Operating Systems Concepts 10th Edition.

## *3.3 Programming project 3*

For this project you will solve problems related to a multithreaded application. Use either C/C++ or Java, on any platform (Linux, macOS, Windows, etc.)

### 3.3.1 The Project

You have to solve **exercise 6.33** at the end of Chapter 6 in the Operating Systems Concepts 10th Edition.

## *3.4 Programming project 4*

For this project you develop an application related to avoiding deadlock. Use either C/C++ or Java, on any platform (Linux, macOS, Windows, etc.)

### 3.4.1 The Project (Banker's Algorithm)

You have to complete Programming Project at the end of Chapter 8 in the Operating Systems Concepts 10th Edition.

## *3.5 Programming project 5*

For this project you develop an application that simulated contiguous memory allocation. Use either C/C++ or Java, on any platform (Linux, macOS, Windows, etc.)

### 3.5.1 The Project (Contiguous Memory Allocation)

You have to complete Programming Project at the end of Chapter 9 in the Operating Systems Concepts 10th Edition.

## *3.6 Requirements for the report*

Your report must contain the following

- Table of contents
- Introduction where you describe what the project was about
- Describe shortly how you solved the programming project
- A screen-shot of the execution of your program
- A conclusion where you summarize what you learned from the project
- Appendix with the source code of the project

## *3.7 Grading Criteria*

Your assignment will be evaluated after you submit a complete report (including table of contents, introduction, conclusion, etc). The following are the grading criteria:

| | |
|---|---|
| Report | (10 points) |
| Programming | |
| Good coding style | (10 points) |
| Good choice of programming solutions | (20 points) |
| Correct functionality of the program | (60 points) |

# 4 Assignment Submission

No handwritten reports will be accepted. You must electronically submit your source file(s) (part B only) and the report via the ***class courses page*** no later than end of *week 13*. Failure to submit all deliverables on time will be penalized (see course outline for the late submission penalty).

# START EARLY!!!!!! Do your own work!!!! If you have questions, come to see me!!!