

## COMP1562 Logbook (Week 5)

### Basic Information

1.1	Student name	<b>Trevor Kiggundu (001001720)</b>
1.2	Who did you work with? Name and/or id	<b>Maruf Hoque (001006731)</b>
1.3	Which lab topic does this document relate to?	Scheduling
1.4	How well do you feel you have done?	I have completed the exercise and am totally satisfied with my work.
1.5	Briefly explain your answer to question 1.4	My group and I were able to successfully follow and complete the tasks. Proof of that is shown below.

### Annotated screenshots demonstrating what you have achieved:

#### EXERCISES:

Exercises a-h required us to run the various algorithm simulations in Workbench in order to monitor the results. Screenshots of the various simulations are shown below:

#### a. FCFS1

Scheduling Algorithms - Introductory

**Process Configuration**

	CPU Intense	Balanced	IO Intense	Runtime (Milliseconds)
Process 1	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	30
Process 2	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	30

**System Configuration**

Number of Processors: 1  
Quantum size: 10 Milliseconds  
IO device latency (constant): 15 Milliseconds  
Priority: Fixed, all processes equal

**Runtime State Display**

RUN: [Empty box]  
READY: [Empty box]  
BLOCKED: [Empty box]  
COMPLETED: [Process 1 icon, Process 2 icon]

**Scheduler Configuration**

☐ Round Robin  
☒ First Come First Served  
☐ Shortest Job First

**Animation Control**

Simulation speed: [Slider from Fastest to Slowest]

**System Statistics**

Elapsed Time (Milliseconds): 60

Balanced processes perform IO every 15 ms of Runtime  
IO-intensive processes perform IO every 7 ms of Runtime

**Runtime Statistics (Milliseconds)**

	Runtime Used	Runtime Remaining	Wait time	IO Time (Blocked)	Total time in system
Process 1	30	0	0	0	30
Process 2	30	0	30	0	60

Free Run Pause Reset Simulation Done

Turn OFF Icon Flashing

## b. SJF 1

## Scheduling Algorithms - Introductory

The screenshot displays the 'Process Scheduler Simulation' interface. It is divided into several functional areas:

- Process Configuration:** Located at the top left, it shows two processes:
  - Process 1:** CPU Intense (selected), Balanced, IO Intense. Runtime: 300 milliseconds.
  - Process 2:** CPU Intense (selected), Balanced, IO Intense. Runtime: 30 milliseconds.
- System Configuration:** Located at the top right, it includes:
  - Number of Processors: 1
  - Quantum size: 10 milliseconds
  - IO device latency (constant): 15 milliseconds
  - Priority: Fixed, all processes equal
- Turn OFF Icon Flashing:** A button located to the right of the System Configuration.
- Runtime State Display:** A central area showing the state of processes:
  - RUN:** A yellow box representing the current state of the processes.
  - READY:** A yellow box representing the queue of processes waiting to be scheduled.
  - BLOCKED:** A yellow box representing processes waiting for I/O.
  - COMPLETED:** A yellow box containing two colored circles (red and blue) representing finished processes.
- Scheduler Configuration:** Located on the right, it shows:
  - Round Robin (selected)
  - First Come First Served
  - Shortest Job First
- Animation Control:** Includes a 'Simulation speed' slider ranging from 'Fastest' to 'Slowest'.
- System Statistics:** Shows 'Elapsed Time (Milliseconds)' as 330.
- Runtime Statistics (Milliseconds):** A table at the bottom left showing the progress of each process:
 

	Runtime Used	Runtime Remaining	Wait time	IO Time (Blocked)	Total time in system
Process 1	300	0	30	0	330
Process 2	30	0	0	0	30
- Control Buttons:** At the bottom right, there are four buttons: 'Free Run', 'Pause', 'Reset Simulation', and 'Done'.

c. FCFS2

## Scheduling Algorithms - Introductory

### Process Configuration

	CPU Intense	Balanced	IO Intense	Runtime (Milliseconds)
<div style="display: flex; align-items: center;"><div style="width: 15px; height: 15px; background-color: red; margin-right: 5px;"></div> Process 1</div>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="text" value="30"/>
<div style="display: flex; align-items: center;"><div style="width: 15px; height: 15px; background-color: yellow; margin-right: 5px;"></div> Process 2</div>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="text" value="30"/>

### System Configuration

Number of Processors: 1

Quantum size: 10 Milliseconds

IO device latency (constant): 15 Milliseconds

Priority: Fixed, all processes equal

Turn OFF  
Icon Flashing

### Runtime State Display

#### RUN

#### BLOCKED

#### READY

#### COMPLETED

### Scheduler Configuration

☐ Round Robin

☒ First Come First Served

☐ Shortest Job First

### Animation Control

Simulation speed

Fastest

Slowest

### System Statistics

Elapsed Time (Milliseconds)

Balanced processes perform IO every 15 ms of Runtime

IO-intensive processes perform IO every 7 ms of Runtime

### Runtime Statistics (Milliseconds)

	Runtime Used	Runtime Remaining	Wait time	IO Time (Blocked)	Total time in system
<div style="display: flex; align-items: center;"><div style="width: 15px; height: 15px; background-color: red; margin-right: 5px;"></div> Process 1</div>	<input type="text" value="30"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="30"/>
<div style="display: flex; align-items: center;"><div style="width: 15px; height: 15px; background-color: yellow; margin-right: 5px;"></div> Process 2</div>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>

Free Run

Pause

Reset Simulation

Done



## Scheduling Algorithms - Introductory

g. RR2

The screenshot displays the "Process Scheduler Simulation" window. It includes several panels:



- Process Configuration**: A table for configuring two processes.
 

	CPU Intense	Balanced	IO Intense	Runtime (Milliseconds)
Process 1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	30
Process 2	<input checked="" type="checkbox"/>	<input type="radio"/>	<input type="radio"/>	30
- System Configuration**:
  - Number of Processors: 1
  - Quantum size: 10 Milliseconds
  - IO device latency (constant): 15 Milliseconds
  - Priority: Fixed, all processes equal
- Turn OFF Icon Flashing**: A button located at the top right.
- Runtime State Display**: A central area with four boxes labeled RUN, READY, BLOCKED, and COMPLETED. The COMPLETED box contains icons for both processes.
- Scheduler Configuration**: Radio buttons for Round Robin (selected), First Come First Served, and Shortest Job First.
- Animation Control**: A slider for "Simulation speed" ranging from Fastest to Slowest.
- System Statistics**: A section showing "Elapsed Time (Milliseconds)" as 60.
- Runtime Statistics (Milliseconds)**: A table summarizing process execution times.
 

	Runtime Used	Runtime Remaining	Wait time	IO Time (Blocked)	Total time in system
Process 1	30	0	20	0	50
Process 2	30	0	30	0	60
- Control Buttons**: At the bottom are buttons for Free Run, Pause, Reset Simulation, and Done.

## Scheduling Algorithms - Introductory

Process Configuration

	CPU Intense	Balanced	IO Intense	Runtime (Milliseconds)
 Process 1	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	60
 Process 2	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	60

System Configuration

Number of Processors: 1

Quantum size: 10 Milliseconds

IO device latency (constant): 15 Milliseconds

Priority: Fixed, all processes equal

Turn OFF Icon Flashing

Runtime State Display

RUN

READY

BLOCKED

COMPLETED

Scheduler Configuration

☒ Round Robin

☐ First Come First Served

☐ Shortest Job First

Animation Control

Simulation speed

Fastest  Slowest

System Statistics



Elapsed Time (Milliseconds)

120

Balanced processes perform IO every 15 ms of Runtime

IO-intensive processes perform IO every 7 ms of Runtime

Runtime Statistics (Milliseconds)

	Runtime Used	Runtime Remaining	Wait time	IO Time (Blocked)	Total time in system
 Process 1	60	0	50	0	110
 Process 2	60	0	60	0	120

Free Run

Pause



Reset Simulation

Done

## h. RR3

### Scheduling Algorithms - Introductory

Process Configuration

	CPU Intense	Balanced	IO Intense	Runtime (Milliseconds)
 Process 1	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	50
 Process 2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	60

System Configuration

Number of Processors: 1

Quantum size: 10 Milliseconds

IO device latency (constant): 15 Milliseconds

Priority: Fixed, all processes equal

Turn OFF Icon Flashing

Runtime State Display

RUN

READY

BLOCKED

COMPLETED

Scheduler Configuration

☒ Round Robin

☐ First Come First Served

☐ Shortest Job First

Animation Control

Simulation speed

Fastest  Slowest

System Statistics



Elapsed Time (Milliseconds)

50

Balanced processes perform IO every 15 ms of Runtime

IO-intensive processes perform IO every 7 ms of Runtime

Runtime Statistics (Milliseconds)

	Runtime Used	Runtime Remaining	Wait time	IO Time (Blocked)	Total time in system
 Process 1	50	0	0	0	50
 Process 2	0	0	0	0	0

Free Run

Pause

Reset Simulation

Done

### **TASKS:**

For the following scenario with four processes please draw Gantt's charts and calculate *average waiting time*  $\underline{t_{AWT}}$  and *average time a process remains in the system*  $\underline{t_{ATT}}$  for the following scheduling algorithms:

#### **SCENARIO:**

Process	Arrival Time (When it arrives in the system) [ms]	Service Time (From arrival to finish) [ms]
P1	0	11
P2	2	8
P3	4	4
P4	6	6

### **TASK 5.1:**

#### **First Comes First Served:**

The FCFS algorithm is shown below. The processes in the algorithm ran completely in order (P1, P2, P3, P4), with the total time in the system increasing as each process was carried out. The calculated average wait time for each process was 10.25 ms, and the average time a process remained in the system was 17.5 ms. The scenario table was unchanged, and the detailed  $\underline{t_{AWT}}$  and  $\underline{t_{ATT}}$  table, code, gantt chart and screenshots are shown below:

#### **$\underline{t_{AWT}}$ and $\underline{t_{ATT}}$ table:**

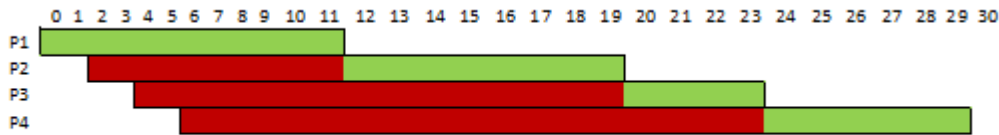
Process	Wait Time ( $\underline{t_{AWT}}$ ) [ms]	Average time a process remains in system ( $\underline{t_{ATT}}$ ) [ms]
P1	0	11
P2	9	17
P3	15	19
P4	17	23
Average:	10.25	17.5

#### **Scriptcheck Code:**

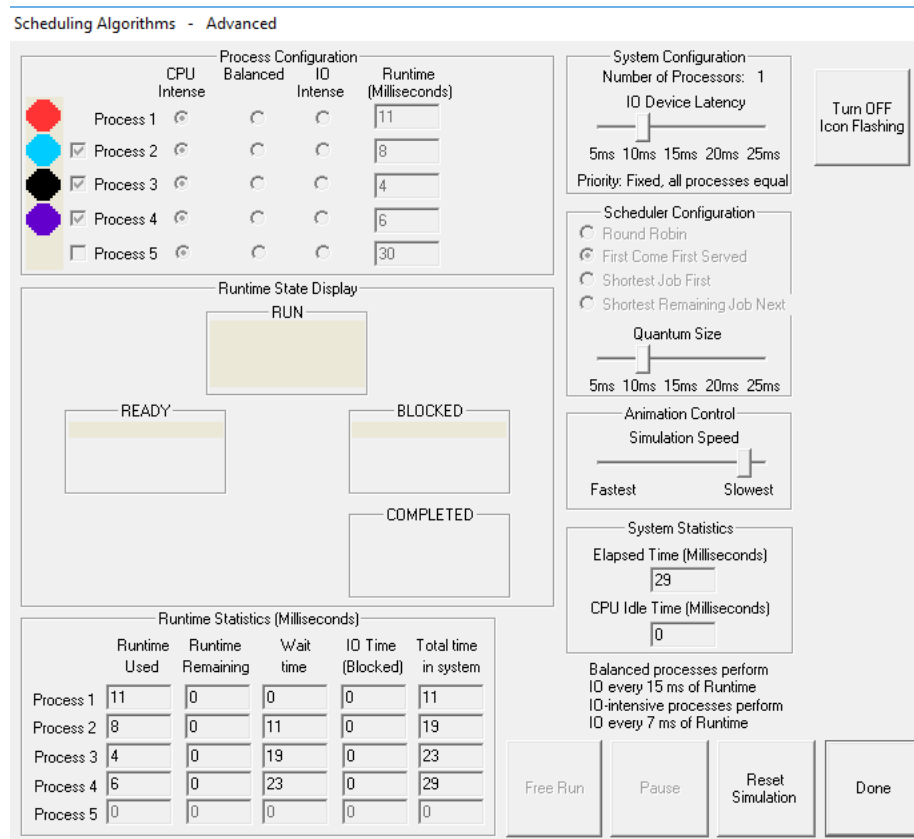
TTTTTTTTTT

```
--WWWWWWWWWWTTTTTTTTT
----WWWWWWWWWWWWWWWWTTTT
-----WWWWWWWWWWWWWWWWTTTTTT
```

Gantt Chart: (x-axis: process, y-axis: time[ms] ):



## Workbench Screenshot:



## TASK 5.2:

### Shortest Job Next:

The SJN algorithm is shown below. The processes in the algorithm ran in a different order than FCFS (P1, P3, P4, P2), with the total time in the system fluctuating as each process was carried out. The calculated average wait time for each process was 8.75 ms, and the average time a process remained in the system was 16 ms. The updated scenario table, detailed  $t_{AWT}$  and  $t_{ATT}$  table, code, gantt chart and screenshots are shown below:

## SCENARIO:

Process	Arrival Time (When it arrives in the system) [ms]	Service Time (From arrival to finish) [ms]
P1	0	11
P3	4	4
P4	6	6
P2	2	8

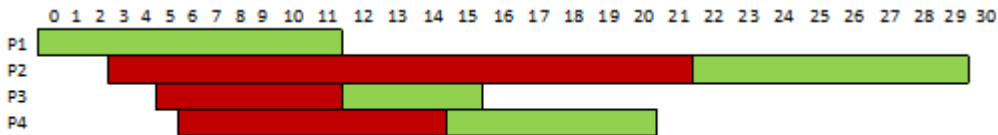
**t<sub>AWT</sub> and t<sub>ATT</sub> table:**

Process	Wait Time ( $\underline{t_{AWT}}$ ) [ms]	Average time a process remains in system ( $\underline{t_{ATT}}$ ) [ms]
P1	0	11
P2	9	15
P3	19	27
P4	7	11
Average:	8.75	16

**Scriptcheck Code:**

```
TTTTTTTTTT
----WWWWWWWrrrr
-----WWWWWWWWWrrrrrr
--WWWWWWWWWWWWWWWWWWWWWWWWWWWWrrrrrrrrrr
```

**Gantt Chart: (x-axis: process, y-axis: time[ms]):**





## Workbench Screenshot:



## Personal Reflection:

This weeks' task was surprisingly easy compared to last weeks'. We were required to familiarize ourselves with the WorkBench program, one which I had never encountered before. However, after running through the pre-task exercises, I was able to confidently use the program. I found it so cool that I could pause and simulate multiple processes and algorithms, mimicking how they would actually run in real time situations. Unlike the previous weeks' we had no problem using scriptcheck to upload out answers, as the input system was fairly straight forward. The tasks required us to learn what we had used in the exercises and apply it to different scenarios, which was easy enough. We had also made gantt charts in previous modules so that was an easy task as well. Calculating the wait and average time in the system was a matter of simple math, and before we knew it, we were finished with week 5 tasks. I thoroughly enjoyed the task and hope the future ones are as easy to understand as this one. I do believe that I have furthered my understanding of operating systems and look forward to continuing so next week.