OPERATING SYSTEMS LAB REPORT

Suyash Gaurav (210010054) & Vivek Gaikwad (210010059)

LAB 4

GOAL

Designing SJF and RR schedulers

(1) compute various performance measures (turnaround time,waiting time, penalty ratio for each process and system averages and system throughput)

(2) analyze the behavior of your schedulers. Your output should include results per process and the system's overall performance

SHORTEST JOB FIRST (SJF)

Explanation of SJF

It is a non-preemptive scheduling method which is used in OS to manage CPU time allocation to various processes. SJF operates on the principle that the CPU should be assigned to the process with the smallest anticipated execution duration.

The priority queue (pq_cpu) is utilized for selecting the process with the shortest CPU burst time during CPU scheduling. The priority queue is implemented using a custom comparator class (Compare) which ensures that processes are ordered first by their CPU burst times and, in case of a tie, by their arrival times.

During the execution, the program continually selects processes from the priority queue for CPU execution.

Processes are added to the priority queue when they become available for execution based on their arrival times. If a process arrives while the CPU is idle, it is immediately added to the priority queue.

As a process runs on the CPU, its remaining CPU burst time is decremented. Upon completion of a CPU burst, processes are checked for any remaining IO burst times. If present, the process is enqueued into a separate queue (q_io) to simulate IO operations.

If the process still has CPU burst time remaining, it is re-inserted into the priority queue to be selected for execution again.

According to SJF, the loop continues until both the CPU and IO queues are empty.

Expected job characteristics

- Non-preemptive: It does not allow for interruptions & a process will continue to execute until it is finished or blocked/suspended state which is gone for I/O.
- Priority based algorithm: Since it uses cpu burst time as priority the process with the shortest burst time will be given highest priority.
- Optimize Throughput: Since the shortest processes are executed first thus overall waiting time for processes is reduced.
- Starvation: SJF may cause very long turn-around times or starvation.
- Job completion: In SJF job completion time must be known earlier, but sometimes it is hard to predict.

Test process data to bring out the suitability of SJF

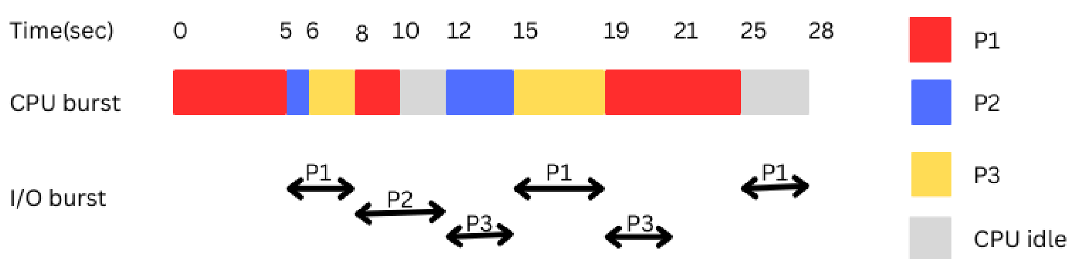
Since the scheduler is non-preemptive it is suitable for prioritizing jobs having short execution time and also they won’t have to undergo starvation.

The time sharing will be according to the execution time.

A sample test case is given below : *suiatable_sjf.dat*

Process	Arrival time	CPU burst	I/O burst	CPU burst	I/O burst	CPU burst	I/O burst	Total CPU burst
P1	0	5	3	2	4	6	3	13
P2	2	1	4	3	-	-	-	4
P3	5	2	3	4	2	-	-	6

Gantt Chart :



Desired output on running above test case :

```
Process P1:
Turnaround Time: 28
Waiting Time: 0
Penalty Ratio: 1.21739
Completion Time: 28
###################################################

Process P2:
Turnaround Time: 13
Waiting Time: 3
Penalty Ratio: 1.625
Completion Time: 15
###################################################

Process P3:
Turnaround Time: 16
Waiting Time: 1
Penalty Ratio: 1.45455
Completion Time: 21
###################################################

Average Turnaround Time: 19
Average Waiting Time: 1.33333
Average Penalty Ratio: 1.43231
Throughput: 0.107143

```

Conclusion :

- SJF is suitable for P2 as it prioritizes the shortest job, leading to efficient CPU utilization and minimizing the turnaround time for P2.
- SJF is suitable for all processes due to short total CPU burst, reducing waiting time.
- SJF is suitable for all processes since it has almost similar penalty ratio for all.

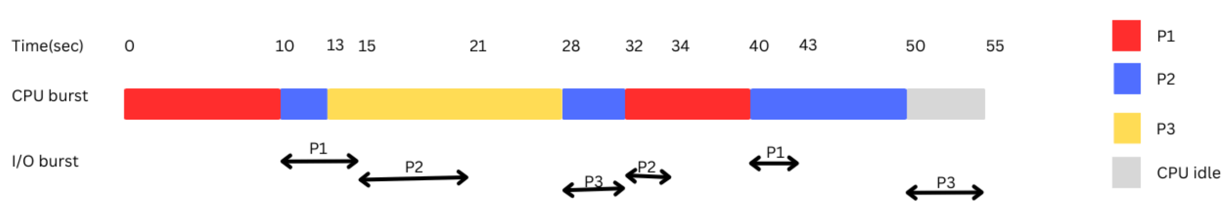
Test process data to bring out the shortcomings of SJF

To highlight the shortcomings of the SJF code, let's create a test case that involves a mix of short and long processes with varying arrival times and burst times. This will help demonstrate how SJF may not always provide optimal results and may lead to increased waiting times for certain processes and starvation.

Below given a test case: *shortcoming_sjf.dat*

Process	Arrival time	CPU burst	I/O burst	CPU burst	I/O burst	Total CPU burst
P1	0	10	5	8	3	18
P2	5	3	6	4	2	7
P3	8	15	4	10	5	25

Gantt Chart :



Desired output on running above test case :

```
Process P1:
Turnaround Time: 43
Waiting Time: 17
Penalty Ratio: 1.65385
Completion Time: 43
#################################################
#

Process P2:
Turnaround Time: 29
Waiting Time: 12
Penalty Ratio: 1.93333
Completion Time: 34
#################################################
#

Process P3:
Turnaround Time: 47
Waiting Time: 13
Penalty Ratio: 1.38235
Completion Time: 55
#################################################
#

Average Turnaround Time: 39.6667
Average Waiting Time: 14
Average Penalty Ratio: 1.65651
Throughput: 0.0545455

```

Conclusion :

- P3, arrives early and prevents shorter processes from getting CPU time correctly. In this case, P2 has to wait until P3 completes its execution, causing a delay.
- Since P3 is executing at the end it has remaining I/O thus causing the CPU to be idle.
- SJF may not always be suitable in scenarios where there is a mix of short(P2) and

long (P1,P3) processes, as it tends to favor the shortest job without considering the overall waiting time of processes.

The analysis of the performance of SJF when run on the test cases

Process1.dat :

```
Process P1:
Turnaround Time: 1097
Waiting Time: 676
Penalty Ratio: 2.6057
Completion Time: 1097
##################################################
#

Process P2:
Turnaround Time: 1436
Waiting Time: 1095
Penalty Ratio: 4.21114
Completion Time: 1438
##################################################
#

Process P3:
Turnaround Time: 1084
Waiting Time: 793
Penalty Ratio: 3.72509
Completion Time: 1087
##################################################
#

Process P4:
Turnaround Time: 573
Waiting Time: 372
Penalty Ratio: 2.85075
Completion Time: 577
##################################################
#

```

```
Process P5:
Turnaround Time: 179
Waiting Time: 166
Penalty Ratio: 13.7692
Completion Time: 184
##################################################
#

Process P6:
Turnaround Time: 99
Waiting Time: 94
Penalty Ratio: 19.8
Completion Time: 105
##################################################
#

Process P7:
Turnaround Time: 477
Waiting Time: 272
Penalty Ratio: 2.32683
Completion Time: 487
##################################################
#

Average Turnaround Time: 706.429
Average Waiting Time: 495.429
Average Penalty Ratio: 7.04125
Throughput: 0.00486787

```

Process2.dat :

```
Process P1:
Turnaround Time: 5
Waiting Time: 0
Penalty Ratio: 1
Completion Time: 5
####################################################
#

Process P2:
Turnaround Time: 83
Waiting Time: 70
Penalty Ratio: 6.38462
Completion Time: 84
####################################################
#

Process P3:
Turnaround Time: 481
Waiting Time: 280
Penalty Ratio: 2.39303
Completion Time: 487
####################################################
#

Process P4:
Turnaround Time: 169
Waiting Time: 156
Penalty Ratio: 13
Completion Time: 192
####################################################
#

Process P5:
Turnaround Time: 813
Waiting Time: 522
Penalty Ratio: 2.79381
Completion Time: 837
####################################################
#

Process P6:
Turnaround Time: 164
Waiting Time: 151
Penalty Ratio: 12.6154
Completion Time: 189
####################################################
#

```

```
Process P7:
Turnaround Time: 935
Waiting Time: 594
Penalty Ratio: 2.74194
Completion Time: 961
####################################################
#

Process P8:
Turnaround Time: 147
Waiting Time: 134
Penalty Ratio: 11.3077
Completion Time: 174
####################################################
#

Process P9:
Turnaround Time: 229
Waiting Time: 192
Penalty Ratio: 6.18919
Completion Time: 257
####################################################
#

Process P10:
Turnaround Time: 142
Waiting Time: 129
Penalty Ratio: 10.9231
Completion Time: 171
####################################################
#

Process P11:
Turnaround Time: 125
Waiting Time: 112
Penalty Ratio: 9.61539
Completion Time: 166
####################################################
#

Process P12:
Turnaround Time: 120
Waiting Time: 107
Penalty Ratio: 9.23077
Completion Time: 153
####################################################
#

Process P13:
Turnaround Time: 103
Waiting Time: 90
Penalty Ratio: 7.92308
Completion Time: 138
####################################################
#

```

```
Process P14:
Turnaround Time: 80
Waiting Time: 67
Penalty Ratio: 6.15385
Completion Time: 120
###################################################

Process P15:
Turnaround Time: 95
Waiting Time: 82
Penalty Ratio: 7.30769
Completion Time: 135
###################################################

Process P16:
Turnaround Time: 75
Waiting Time: 62
Penalty Ratio: 5.76923
Completion Time: 117
###################################################

Process P17:
Turnaround Time: 59
Waiting Time: 46
Penalty Ratio: 4.53846
Completion Time: 102
###################################################

Process P18:
Turnaround Time: 54
Waiting Time: 41
Penalty Ratio: 4.15385
Completion Time: 99
###################################################

Average Turnaround Time: 215.5
Average Waiting Time: 157.5
Average Penalty Ratio: 6.89117
Throughput: 0.0187305

```

Process3.dat :

```
Process P1:
Turnaround Time: 593
Waiting Time: 365
Penalty Ratio: 2.60088
Completion Time: 593
###################################################

Process P2:
Turnaround Time: 1336
Waiting Time: 1042
Penalty Ratio: 4.54422
Completion Time: 1338
###################################################

Process P3:
Turnaround Time: 1378
Waiting Time: 1074
Penalty Ratio: 4.53289
Completion Time: 1383
###################################################

Process P4:
Turnaround Time: 315
Waiting Time: 147
Penalty Ratio: 1.875
Completion Time: 323
###################################################

Process P5:
Turnaround Time: 586
Waiting Time: 417
Penalty Ratio: 3.46746
Completion Time: 598
###################################################

Process P6:
Turnaround Time: 1799
Waiting Time: 1417
Penalty Ratio: 4.70942
Completion Time: 1819
###################################################

```

```
Process P7:
Turnaround Time: 973
Waiting Time: 737
Penalty Ratio: 4.12288
Completion Time: 1003
###################################################

Process P8:
Turnaround Time: 51
Waiting Time: 45
Penalty Ratio: 8.5
Completion Time: 86
###################################################

Process P9:
Turnaround Time: 167
Waiting Time: 107
Penalty Ratio: 2.78333
Completion Time: 203
###################################################

Process P10:
Turnaround Time: 164
Waiting Time: 64
Penalty Ratio: 1.64
Completion Time: 201
###################################################

Process P11:
Turnaround Time: 63
Waiting Time: 54
Penalty Ratio: 7
Completion Time: 101
###################################################

Process P12:
Turnaround Time: 2083
Waiting Time: 1694
Penalty Ratio: 5.35476
Completion Time: 2123
###################################################

Average Turnaround Time: 792.333
Average Waiting Time: 596.917
Average Penalty Ratio: 4.2609
Throughput: 0.00565238

```

How to run code?

Open a terminal or command prompt and navigate to the directory where your ***SJF.cpp*** file is located.

```
suyas@Z-Sparrow:/mnt/c/Users/suyas/OneDrive/Documents/Minix3/LAB_4$ g++ SJF.cpp -o sjf
```

Once the compilation is successful, you can run the compiled program. Our program expects a **command-line argument in the form of a test case file** (e.g., *test_cases/process1.dat*).

```
suyas@Z-Sparrow:/mnt/c/Users/suyas/OneDrive/Documents/Minix3/LAB_4$ ./sjf test_cases/process1.dat
```

The program will print the results to a file named ***SJF_output.txt*** which is in the ***Results*** folder.

ROUND ROBIN (RR)

Explanation of SJF

It is a preemption-based approach that allocates CPU time to processes in a cyclic manner. This algorithm maintains a ready queue, and processes are scheduled for execution based on the specified time slice. If a process's CPU burst is not completed within the time slice, it is moved to the back of the queue to allow other processes to execute.

The code uses a queue to represent the ready queue and another queue for handling I/O bursts. Processes are scheduled in a time-sliced manner until all processes complete their execution.

The algorithm ensures fairness by giving each process a turn to execute, but it may lead to higher turnaround times for processes with longer burst times.

Expected job characteristics:

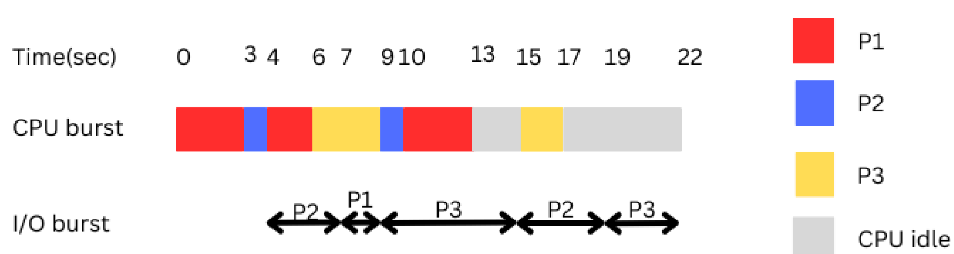
- ➢ Preemptive: A running process can be interrupted before its full time slice is consumed, allowing other processes in the ready queue to get a chance to execute.
- ➢ Starvation-Free: Round Robin is designed to be starvation-free, ensuring that all processes get a fair share of the CPU. No process is left waiting indefinitely, as each process gets a chance to execute within a fixed time SLICE.
- ➢ Fairness: The algorithm provides fairness by ensuring that every process receives an equal share of CPU time.
- ➢ Low throughput : The Round Robin algorithm might have lower throughput, especially when using a large time quantum. This happens because processes may not switch quickly, causing the CPU to not be fully used.
- ➢ Context Switches: Round Robin scheduling involves frequent context switches, as processes are preempted after their time quantum expires.

Test process data to bring out the suitability of RR

A sample test case is given below with time slice 3 sec: *suitable_rr.dat*

Process	Arrival time	CPU burst	I/O burst	CPU burst	I/O burst
P1	0	5	2	3	-
P2	2	1	3	1	4
P3	4	3	6	2	3

Gantt Chart :



Desired output on running above test case :

```
Process P1:
Turnaround Time: 13
Waiting Time: 2
Penalty Ratio: 1.3
Completion Time: 13
##################################################
#

Process P2:
Turnaround Time: 17
Waiting Time: 3
Penalty Ratio: 1.88889
Completion Time: 19
##################################################
#

Process P3:
Turnaround Time: 18
Waiting Time: 2
Penalty Ratio: 1.28571
Completion Time: 22
##################################################
#

Average Turnaround Time: 16
Average Waiting Time: 2.33333
Average Penalty Ratio: 1.49153
Throughput: 0.136364

```

Conclusion :

- Each process gets a time slice to execute its CPU burst, preventing any single process from occupying the CPU.
- This responsiveness is shown in the quick running of P1 and P2 on CPU, which have shorter initial CPU burst times.
- Shorter processes, like P1 and P2, get opportunities to execute within their time slices despite the presence of a longer process (P3).

Test process data to bring out the shortcomings of RR:

Below is given a test example to show shortcomings of RR named *shortcoming_rr.dat* with *time slice* 7.

Process	Arrival time	CPU burst	I/O burst	CPU burst	I/O burst
P1	0	10	5	8	4
P2	2	5	3	7	2
P3	4	12	4	6	6

Conclusion :

- The small time quantum of 7 ms leads to frequent context switches, which can increase overhead.
- Processes are often interrupted before completing their CPU bursts, leading to longer waiting times.
- P1 experiences interruptions in its CPU burst due to the short time quantum. It gets interrupted after using only 7 sec of its initial 10 ms burst.
- P1 is frequently context-switched with other processes, leading to additional overhead and a fragmented execution pattern.

Desired output on running above test case :

```
Process P1:
Turnaround Time: 57
Waiting Time: 25
Penalty Ratio: 2.11111
Completion Time: 57
##################################################

Process P2:
Turnaround Time: 34
Waiting Time: 17
Penalty Ratio: 2
Completion Time: 36
##################################################

Process P3:
Turnaround Time: 49
Waiting Time: 21
Penalty Ratio: 1.75
Completion Time: 53
##################################################

Average Turnaround Time: 46.6667
Average Waiting Time: 21
Average Penalty Ratio: 1.9537
Throughput: 0.0526316
```

The analysis of the performance of SJF when run on the test cases

Process1.dat : *timeslice 3*

```
Process P1:
Turnaround Time: 1430
Waiting Time: 1009
Penalty Ratio: 3.39667
Completion Time: 1430
############################################################
#

Process P2:
Turnaround Time: 1355
Waiting Time: 1014
Penalty Ratio: 3.97361
Completion Time: 1357
############################################################
#

Process P3:
Turnaround Time: 1261
Waiting Time: 970
Penalty Ratio: 4.33333
Completion Time: 1264
############################################################
#

Process P4:
Turnaround Time: 990
Waiting Time: 789
Penalty Ratio: 4.92537
Completion Time: 994
############################################################
#

Process P5:
Turnaround Time: 57
Waiting Time: 44
Penalty Ratio: 4.38462
Completion Time: 62
############################################################
#

Process P6:
Turnaround Time: 32
Waiting Time: 27
Penalty Ratio: 6.4
Completion Time: 38
############################################################
#

Process P7:
Turnaround Time: 999
Waiting Time: 794
Penalty Ratio: 4.87317
Completion Time: 1009
############################################################
#

Average Turnaround Time: 874.857
Average Waiting Time: 663.857
Average Penalty Ratio: 4.6124
Throughput: 0.00489511
```

Process2.dat : *timeslice 3*

```
Process P1:
Turnaround Time: 8
Waiting Time: 3
Penalty Ratio: 1.6
Completion Time: 8
##################################################
#

Process P2:
Turnaround Time: 22
Waiting Time: 9
Penalty Ratio: 1.69231
Completion Time: 23
##################################################
#

Process P3:
Turnaround Time: 711
Waiting Time: 510
Penalty Ratio: 3.53731
Completion Time: 717
##################################################
#

Process P4:
Turnaround Time: 79
Waiting Time: 66
Penalty Ratio: 6.07692
Completion Time: 102
##################################################
#

```

```
Process P5:
Turnaround Time: 889
Waiting Time: 598
Penalty Ratio: 3.05498
Completion Time: 913
##################################################
#

Process P6:
Turnaround Time: 92
Waiting Time: 79
Penalty Ratio: 7.07692
Completion Time: 117
##################################################
#

Process P7:
Turnaround Time: 933
Waiting Time: 592
Penalty Ratio: 2.73607
Completion Time: 959
##################################################
#

Process P8:
Turnaround Time: 105
Waiting Time: 92
Penalty Ratio: 8.07692
Completion Time: 132
##################################################
#

Process P9:
Turnaround Time: 259
Waiting Time: 222
Penalty Ratio: 7
Completion Time: 287
##################################################
#

```

```
Process P10:
Turnaround Time: 118
Waiting Time: 105
Penalty Ratio: 9.07692
Completion Time: 147
###################################################

Process P11:
Turnaround Time: 119
Waiting Time: 106
Penalty Ratio: 9.15385
Completion Time: 150
###################################################

Process P12:
Turnaround Time: 123
Waiting Time: 110
Penalty Ratio: 9.46154
Completion Time: 156
###################################################

Process P13:
Turnaround Time: 124
Waiting Time: 111
Penalty Ratio: 9.53846
Completion Time: 159
###################################################

Process P14:
Turnaround Time: 122
Waiting Time: 109
Penalty Ratio: 9.38461
Completion Time: 162
###################################################

```

```
Process P15:
Turnaround Time: 128
Waiting Time: 115
Penalty Ratio: 9.84615
Completion Time: 168
###################################################

Process P16:
Turnaround Time: 129
Waiting Time: 116
Penalty Ratio: 9.92308
Completion Time: 171
###################################################

Process P17:
Turnaround Time: 134
Waiting Time: 121
Penalty Ratio: 10.3077
Completion Time: 177
###################################################

Process P18:
Turnaround Time: 138
Waiting Time: 125
Penalty Ratio: 10.6154
Completion Time: 183
###################################################

Average Turnaround Time: 235.167
Average Waiting Time: 177.167
Average Penalty Ratio: 7.11995
Throughput: 0.0187696

```

Process3.dat : *timeslice 3*

```
Process P1:
Turnaround Time: 1599
Waiting Time: 1334
Penalty Ratio: 7.01316
Completion Time: 1599
##################################################
#

Process P2:
Turnaround Time: 1922
Waiting Time: 1628
Penalty Ratio: 6.53742
Completion Time: 1924
##################################################
#

Process P3:
Turnaround Time: 1888
Waiting Time: 1584
Penalty Ratio: 6.21053
Completion Time: 1893
##################################################
#

Process P4:
Turnaround Time: 1355
Waiting Time: 1187
Penalty Ratio: 8.06548
Completion Time: 1363
##################################################
#

Process P5:
Turnaround Time: 1324
Waiting Time: 1119
Penalty Ratio: 7.83432
Completion Time: 1336
##################################################
#

Process P6:
Turnaround Time: 2069
Waiting Time: 1687
Penalty Ratio: 5.41623
Completion Time: 2089
##################################################
#

```

```
Process P7:
Turnaround Time: 1638
Waiting Time: 1388
Penalty Ratio: 6.94068
Completion Time: 1668
##################################################
#

Process P8:
Turnaround Time: 58
Waiting Time: 62
Penalty Ratio: 9.66667
Completion Time: 93
##################################################
#

Process P9:
Turnaround Time: 163
Waiting Time: 103
Penalty Ratio: 2.71667
Completion Time: 199
##################################################
#

Process P10:
Turnaround Time: 886
Waiting Time: 786
Penalty Ratio: 8.86
Completion Time: 923
##################################################
#

Process P11:
Turnaround Time: 100
Waiting Time: 91
Penalty Ratio: 11.1111
Completion Time: 138
##################################################
#

Process P12:
Turnaround Time: 2043
Waiting Time: 1654
Penalty Ratio: 5.25193
Completion Time: 2083
##################################################
#

Average Turnaround Time: 1253.75
Average Waiting Time: 1051.08
Average Penalty Ratio: 7.13535
Throughput: 0.00574438

```

How to run code?

Open a terminal and navigate to the directory where your ***rr.cpp*** file is located.

```
suyas@Z-Sparrow:/mnt/c/Users/suyas/OneDrive/Documents/Minix3/LAB_4$ g++ RR.cpp -o rr
```

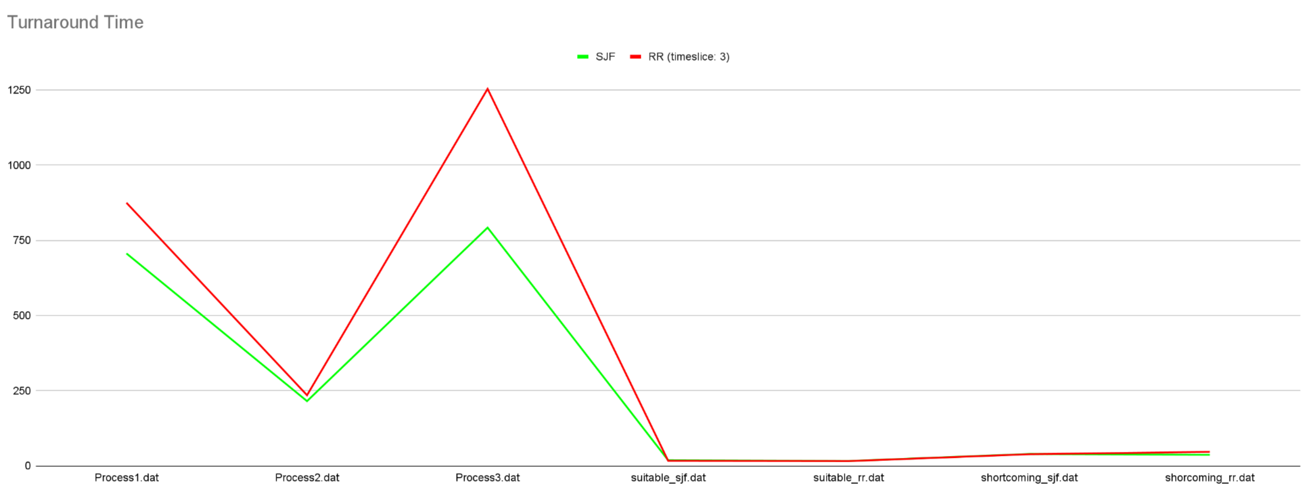
Once the compilation is successful, you can run the compiled program. Our program expects **two command-line arguments in the form of a test case file and followed by timeslice** (e.g., *test_cases/process1.dat 3*).

```
suyas@Z-Sparrow:/mnt/c/Users/suyas/OneDrive/Documents/Minix3/LAB_4$ ./rr test_cases/process1.dat 3
```

The program will print the results to a file named ***RR_output.txt*** which is in the ***Results*** folder.

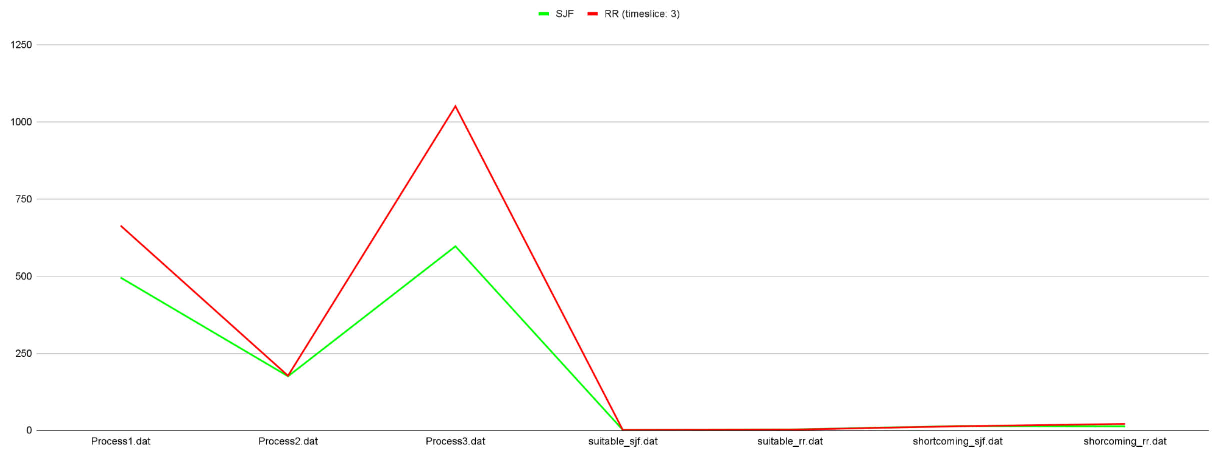
Graphs capturing the variations in performance :

Turnaround Time SJF V/S RR



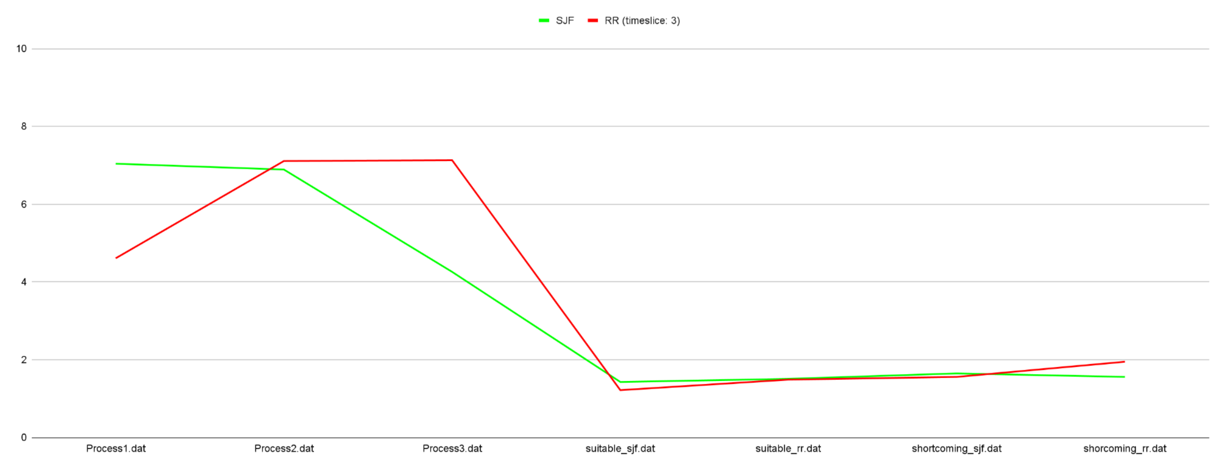
The green line for SJF generally shows lower turnaround times compared to the red line for RR, indicating better performance of SJF.

Waiting time SJF v/s RR



The SJF algorithm performs best on “Process3.dat”, while the RR algorithm performs consistently across all processes. Both algorithms perform equally well on the processes after “suitable_sjf/rr.dat”.

Penalty ratio SJF v/s RR

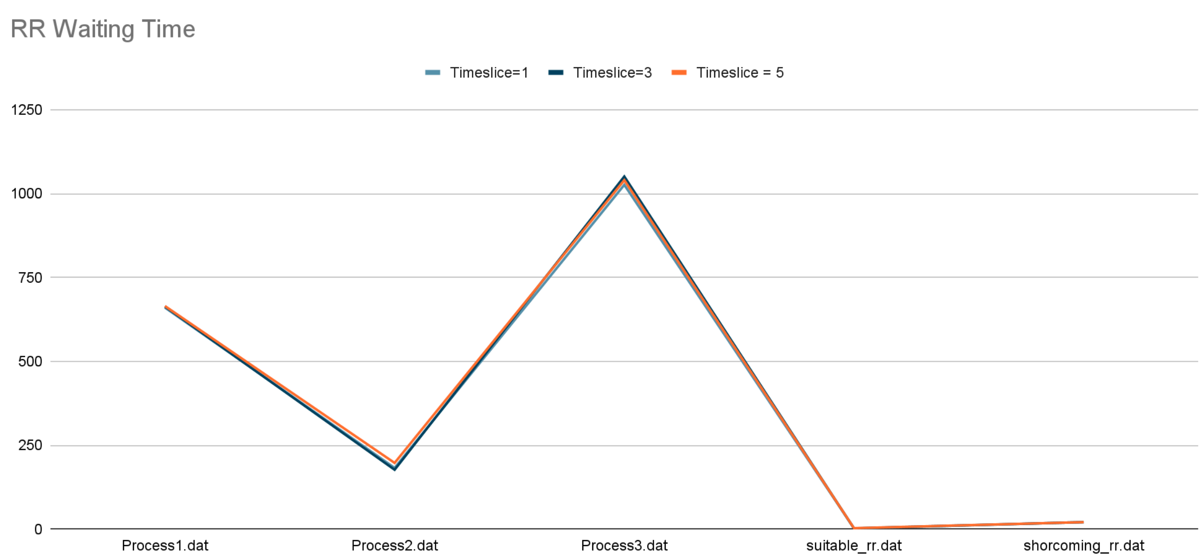


RR shows more fluctuation in penalty ratio compared to SJF as processes change. However, both algorithms show a decline in penalty ratio after Process3.dat.

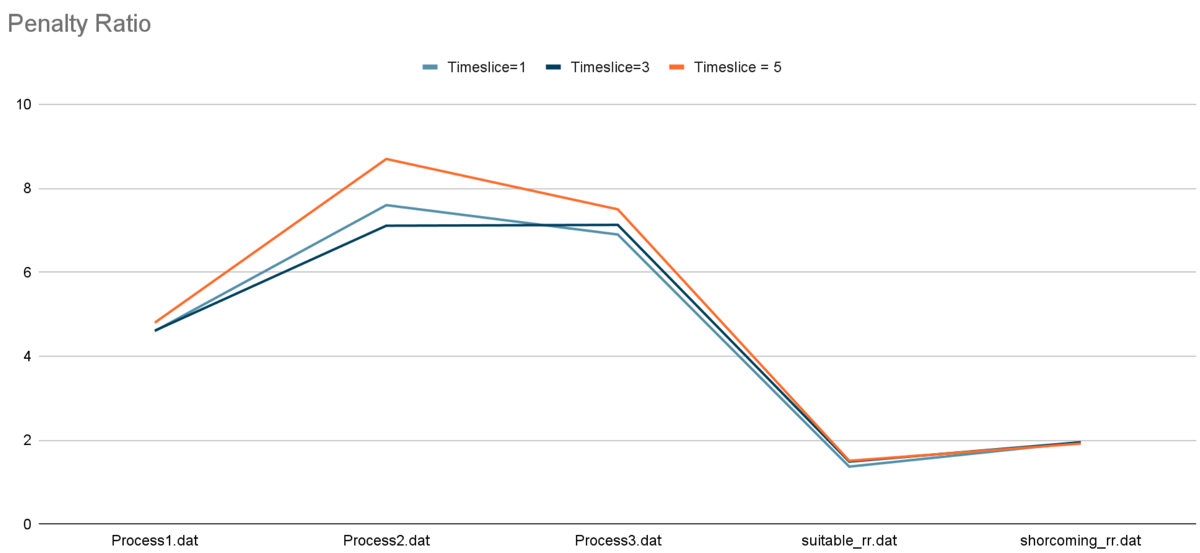
Variation of time slice of RR for process1.dat

Turnaround time

Waiting Time



Penalty Ratio


Penalty Ratio
Timeslice=1
Timeslice=3
Timeslice = 5
10
8
6
4
2
0
Process1.dat
Process2.dat
Process3.dat
suitable_rr.dat
shorcoming_rr.dat