الجامعة الإسلامية العالمية ماليزيا

INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

يونيۏرسيتي اسلام انتارابڠسا مليسيا

Garden of Knowledge and Virtue

SEM2 CSC 3401 Section 5 OPERATING SYSTEMS

SEMESTER 2, 2019/2020

CLASS SCHEDULING

GROUP NAME: CAYLEY

PREPARED BY:

NAME	MATRIC NO.
MAISARAH BINTI HAMZAH	1810325
NUR MAISARA BINTI AHMAD RASID	1818356
NOR NABILAH SHUHADA BINTI MOHD KAMAL	1817510
SITI NUR AISHAH BINTI SULAIMAN	1816454

LECTURER'S NAME : DR NORLIA YUSOF

TABLE OF CONTENT

1.0 INTRODUCTION

Central Processing Unit (CPU) Scheduling is a process made to finish the work on time. In a multiprogramming system, CPU scheduling is needed to manage both Input/Output (IO) and CPU time. While these are the core of computer processing, there are many algorithms available to execute the scheduling process. In this project, we are testing out three algorithms of CPU scheduling that are First Come First Serve (FCFS), Shortest Job First (SJF), and Priority Scheduling and perform an analysis to differentiate between these algorithms in terms of CPU key processes..

2.0 CONSIDERATION

The scheduling algorithms that we considered in this project are First Come First Serve (FCFS), Shortest Job First (SJF), and Priority Scheduling. All of the algorithms that we considered are non-preemptive. We applied processes with the same input such as burst time, priority, and arrival time to evaluate the average waiting time and average turnaround time. Then, with the results, we analyse which one of those three (FCFS scheduling algorithm, SJF scheduling algorithm, and Priority Scheduling) is the better algorithm and optimum best class scheduling.

3.0 ANALYSIS

3.1 INPUT

We initialize the same input in an array proc[] by using the structure name Course for the three algorithms which is given as the following:

```
struct Course
{
   int ccode;      // course code
   int duration;       // class duration
   int priority;  //priority
   int arrival_time;   //prefered arrival time
};

Course proc[] = {{ccode, duration, priority, arrival_time},...};
Course proc[] = {{2201,3,2,0}, {3401, 5, 6,2}, {1103,4,3,1}, {2302,2,5,4},{2602,9,7,6},{3102,4,4,5},{2011,10,10,7}};
```

Course Code	Burst Time	Priority	Arrival Time
2201	3	2	0
3401	5	6	2
1103	4	3	1
2302	2	5	4
2602	9	7	6
3102	4	4	5
2011	10	10	7

Table 1 Initialized input of the algorithms

3.2 OUTPUT

The outputs that we get from FCFS, Priority and SJF scheduling algorithms are the order of scheduling of the course, the attributes of each course code, the average waiting time and the average turnaround time which are given as the following:

i) FCFS

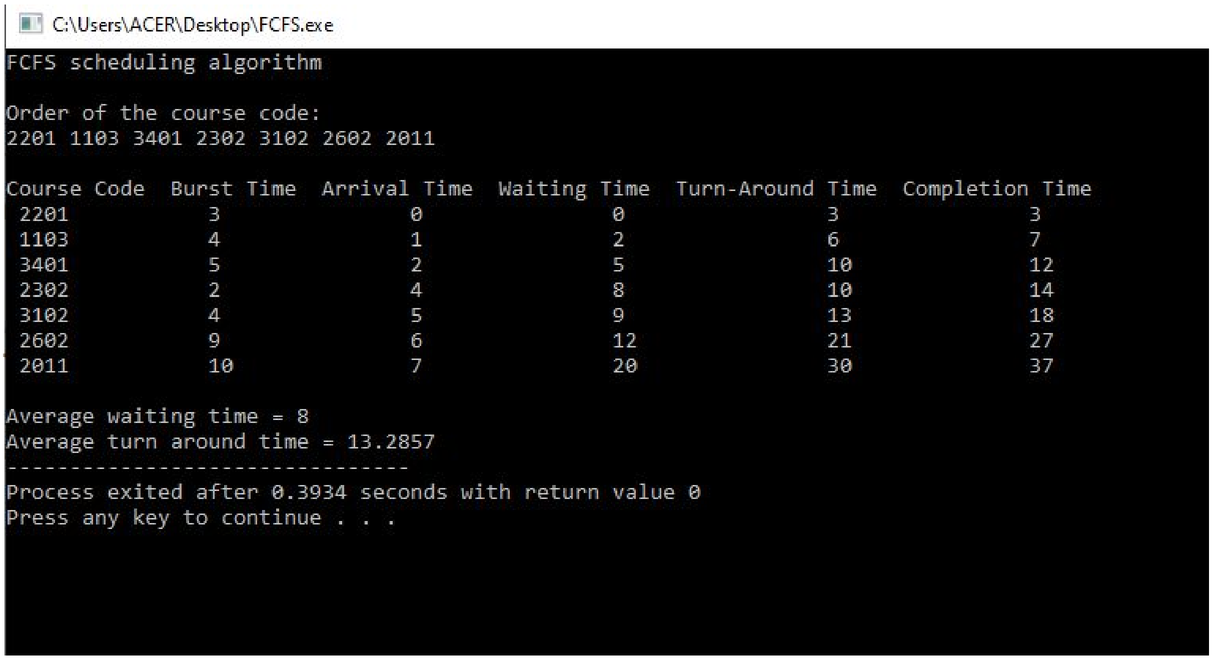
Order of the course code ---> 2201, 1103, 3401, 2302, 3102, 2602, 2011

Course Code	Burst Time	Arrival Time	Waiting Time	Turnaround Time	Completion Time

2201	3	0	0	3	3
1103	4	1	2	6	7
3401	5	2	5	10	12
2302	2	4	8	10	14
3102	4	5	9	13	18
2602	9	6	12	21	27
2011	10	7	20	30	37

Average waiting time = 8

Average turnaround time = 13.2857



C:\Users\ACER\Desktop\FCFS.exe

```
FCFS scheduling algorithm

Order of the course code:
2201 1103 3401 2302 3102 2602 2011

Course Code  Burst Time  Arrival Time  Waiting Time  Turn-Around Time  Completion Time
 2201          3            0             0             3                3
 1103          4            1             2             6                7
 3401          5            2             5             10               12
 2302          2            4             8             10               14
 3102          4            5             9             13               18
 2602          9            6             12            21               27
 2011          10           7             20            30               37

Average waiting time = 8
Average turn around time = 13.2857
--------------------------------
Process exited after 0.3934 seconds with return value 0
Press any key to continue . . .
```

Fig. 1 The output of FCFS

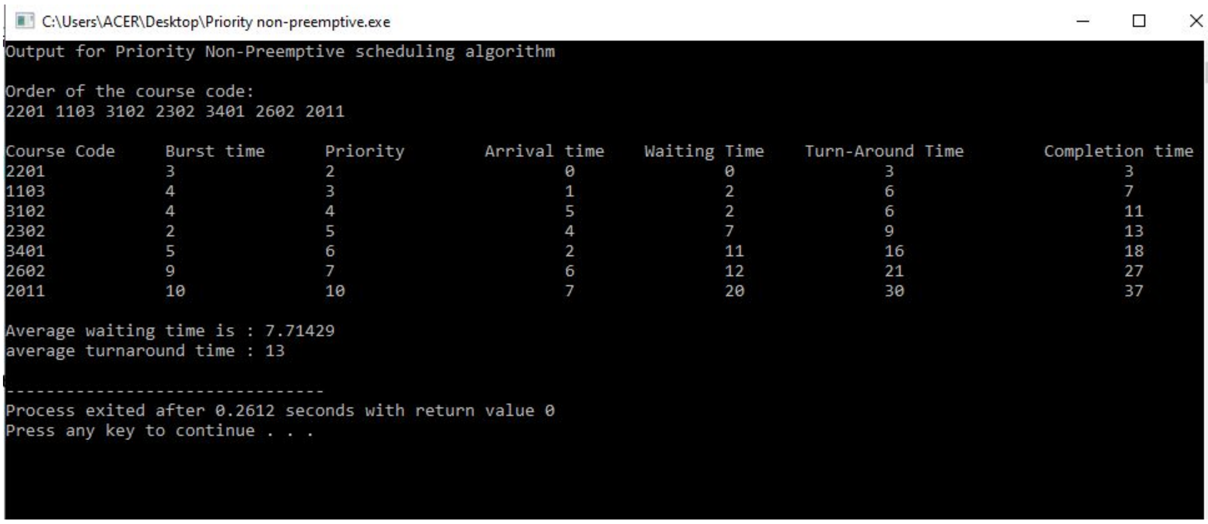
ii) Priority(non-preemptive)

Order of the course code ---> 2201, 1103, 3102, 2302, 3401, 2602, 2011

Course Code	Burst Time	Priority	Arrival Time	Waiting Time	Turnaround Time	Completion Time
2201	3	2	0	0	3	3
1103	4	3	1	2	6	7
3102	4	4	5	2	6	11
2302	2	5	4	7	9	13
3401	5	6	2	11	16	18
2602	9	7	6	12	21	27
2011	10	10	7	20	30	37

Average waiting time = 7.71429

Average turnaround time = 13



```
Output for Priority Non-Preemptive scheduling algorithm

Order of the course code:
2201 1103 3102 2302 3401 2602 2011

Course Code     Burst time      Priority        Arrival time    Waiting Time    Turn-Around Time        Completion time
2201            3               2                       0               0               3                       3
1103            4               3                       1               2               6                       7
3102            4               4                       5               2               6                       11
2302            2               5                       4               7               9                       13
3401            5               6                       2               11              16                      18
2602            9               7                       6               12              21                      27
2011            10              10                      7               20              30                      37

Average waiting time is : 7.71429
average turnaround time : 13

--------------------------------
Process exited after 0.2612 seconds with return value 0
Press any key to continue . . .
```

Fig. 2 The output of Priority non-preemptive scheduling

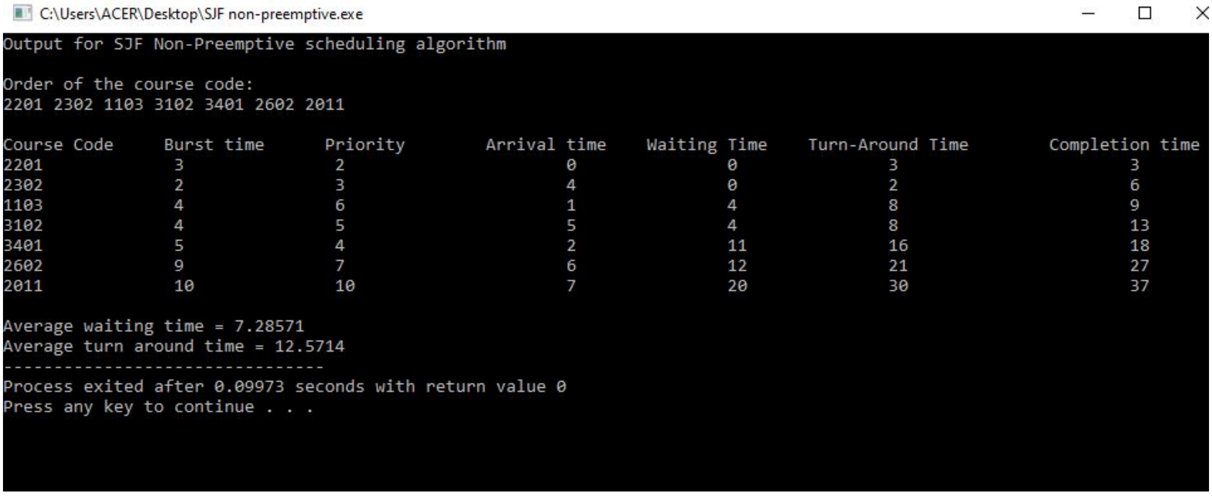
iii) SJF

Order of the course code ---> 2201, 2302, 1103, 3102, 3401, 2602, 2011

Course Code	**Burst Time**	**Priority**	**Arrival Time**	**Waiting Time**	**Turnaround Time**	**Completion Time**
2201	3	2	0	0	3	3
2302	2	5	4	0	2	6
1103	4	3	1	4	8	9
3102	4	4	5	4	8	13
3401	5	6	2	11	16	18
2602	9	7	6	12	21	27
2011	10	10	7	20	30	37

Average waiting time = 7.28571

Average turnaround time = 12.5714



```
Output for SJF Non-Preemptive scheduling algorithm

Order of the course code:
2201 2302 1103 3102 3401 2602 2011

Course Code     Burst time      Priority        Arrival time    Waiting Time    Turn-Around Time        Completion time
2201             3               2                      0               0               3                       3
2302             2               3                      4               0               2                       6
1103             4               6                      1               4               8                       9
3102             4               5                      5               4               8                       13
3401             5               4                      2               11              16                      18
2602             9               7                      6               12              21                      27
2011             10              10                     7               20              30                      37

Average waiting time = 7.28571
Average turn around time = 12.5714
--------------------------------
Process exited after 0.09973 seconds with return value 0
Press any key to continue . . .
```

Fig. 3 The output of SJF scheduling

3.3 COMPARISON CPU SCHEDULING

CHARACTERISTICS	FCFS	SJF	PRIORITY
1) Output order	2201, 1103, 3401, 2302, 3102, 2602, 2011	2201, 2302, 1103, 3102, 3401, 2602, 2011	2201, 1103, 3102, 2302, 3401, 2602, 2011.
2) Waiting time	The average waiting time (8) is the largest among the other two algorithms thus results in a worst performance.	The average waiting time is (7.28571) is the smallest among the other two algorithms.	The average waiting time (7.71429) is smaller than FCFS scheduling algorithm but bigger than SJF scheduling algorithm.
3) Turnaround time	The average turnaround time (13.2857) is the longest among the other two algorithms.	The average turnaround time (12.5714) is the shortest among the other two algorithms.	The average turnaround time (13) is shorter than FCFS scheduling algorithm but bigger than SJF scheduling algorithm.
4) Complexity	FCFS Scheduling is easy and simple to understand. The order of outputs can be easily known according to the input order.	SJF Scheduling is difficult and complex to understand as it is impossible to implement in interactive systems where required CPU time is not known.	The algorithm is quite difficult to understand.
5) Allocation	Job are executed on a first come, first serve	The process holds it till it reaches a waiting	Based on the priority. Higher priority job

	basis.	state or terminated once the CPU cycle is allocated to process,	can execute first.

4.0 CONCLUSION

As a conclusion of our research, we can say that FCFS is the best algorithm and best optimum class scheduling as the SJF is better in minimizing the average waiting time thus resulting in a more effective and higher performance compared to the other two algorithms. The SJF is better if the process comes to a processor simultaneously and can minimize the waiting time. SJF can also lead to higher effectiveness to the systems due to the lower average waiting time. However, it may happen that long processes may never be processed by the system and may remain in the queue for a long period. As for the FCFS, it is easy to understand and implement, however it is poor in performance as their average waiting time is the highest than the rest class scheduling thus can leads to convoy effect. Convoy effect will eventually result in lower device or CPU utilization and lower efficiency. As for the priority scheduling, it is best suited for real-time operating systems, however, it may happen that a low priority process would keep waiting for an indefinite time and never get executed as the systems keep executing the high priority processes.

5.0 REFERENCES

1. UKEssays. (November 2018). A Comparison of CPU Scheduling. Retrieved from https://www.ukessays.com/essays/computer-science/a-comparison-of-cpu-scheduling.php?vref=1

2. Mohitg593Check out this Author's contributed articles., Mohitg593, & Check out this Author's contributed articles. (2020, June 09). Difference between FCFS and Priority CPU scheduling. Retrieved from https://www.geeksforgeeks.org/difference-between-fcfs-and-priority-cpu-scheduling/

3. Mohitg593Check out this Authors contributed articles., Mohitg593, & Check out this Authors contributed articles. (2020, May 04). Difference between FCFS and SJF CPU scheduling algorithms. Retrieved from https://www.geeksforgeeks.org/difference-between-fcfs-and-sjf-cpu-scheduling-algorithms/?ref=rp

4. Shortest Job First (SJF): Preemptive, Non-Preemptive Example. (n.d.). Retrieved from https://www.guru99.com/shortest-job-first-sjf-scheduling.html#:~:text=Shortest Job First (SJF) is,for other processes awaiting execution.
5. First Come First Serve(FCFS) Scheduling Algorithm | Studytonight. (n.d.). Retrieved August 17, 2020, from https://www.studytonight.com/operating-system/first-come-first-serve

6. Priority CPU Scheduling with different arrival time - Set 2 - GeeksforGeeks. (n.d.). Retrieved August 17, 2020, from https://www.geeksforgeeks.org/priority-cpu-scheduling-with-different-arrival-time-set-2/
7. Shortest Job First Scheduling Algorithm | Studytonight. (n.d.). Retrieved August 17, 2020, from https://www.studytonight.com/operating-system/shortest-job-first