SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v7i4.11665 p-ISSN : 2541-044X

Simulation of Priority Round-Robin Scheduling Algorithm

Tri Dharma Putra[1], **Rakhmat Purnomo**[2*]
Universitas Bhayangkara Jakarta Raya
rakhmat.purnomo@dsn.ubharajaya.ac.id

Submitted : Aug 9, 2022 | **Accepted** : Aug 18, 2022 | **Published** : Oct 3, 2022

Abstract: In this journal, simulation of priority round robin scheduling algorithm is presented. To imitate the processes of operating system operation, simulation can be used. By simulation, model is used, namely models that represent the characteristics or behaviour of systems. Process scheduling is one important operation in operating system. OS-SIM can be used to model and simulate the operations of process scheduling. Some scheduling algorithms are available in modern operating systems, like First come First Serve (FCFS), Shortest Job First (SJF), Round Robin (RR), Priority Scheduling or combination of these algorithms. One important scheduling algorithm for real-time or embedded system is priority round robin scheduling algorithm. Priority round robin scheduling algorithm is a preemptive algorithm. Each process is given time quantum. Each process has a priority. Here time quantum 3 is given. The higher the time quantum, the more the context switching. By the use of OS-SIM, simulation can be understood easily and thoroughly. The statistics, will be calculated automatically by the system by the simulator, like the number of context switching, average waiting time, average turn around time, and average responds time. With one example, by using quantum=3. The average turn around time is 18.25 ms. The Average Waiting Time is 12 ms. The Average Responds time is 2.75 ms. The total burst time is 25 ms.

Keywords: Context Switching, OS-SIM, Priority Round Robin Scheduling Algorithm, Process Scheduling, Quantum

INTRODUCTION

The role of Operating System (OS) is as an interface between computer hardware and user. Operating Systems acts as a resource manager. Process scheduling in operating system is one important and fundamental design (Putra, 2020) (Putra & Purnomo, 2022) (Tri Dharma Putra, 2021). Process scheduling is a set of rules, mechanism, and policies that govern the allocated resources to many processes and finishing the scheduling. Process scheduling is a method of how to manage many queues of process to make delay as minimum as possible. Also to make the performance of the system optimal (Dhruv, 2019). The scheduler is operating system's module that arranging the policy of scheduling. The main purpose of scheduling is to make performance of the system optimal that match with the criteria set by designer (Vinay Kumar Reddy & Aakash, 2021) (Putra, 2022).

The prime concept of scheduling in real-time and multiprocessing operating systems design is by arranging switching among the processes of CPU. Priority round robin scheduling algorithm is a well known concept of algorithm in CPU processing (Putra & Purnomo, 2022) (Putra & Purnomo, 2021). A scheduler module is a tool to arrange the policy of scheduling. The main purpose of the scheduler is to optimize the performance of system, that is in line with the criteria set by system designer (m. LaxmiJeevani, T.S.P. Madhuri, 2018). Operating System Concepts Simulator, OS-SIM, is an application to simulate the concept of operating system's scheduling and support the learning process of students (Alexmazinho, 2022) .

There are many existing process scheduling algorithms in the real world. Some are just theoretical some can be implemented. Here are several well known scheduling algorithms: The Shortest Job First (SJF) algorithm. System will execute the process which has the shortest burst time first. In this algorithm, starvation can occur. With this algorithm, system can get average waiting time which is minimum. If compared to other scheduling

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11665
e-ISSN : 2541-2019
p-ISSN : 2541-044X

algorithms (Asma Joshita Trisha, 2019). In priority scheduling, priority is given to each process. The process which have the same priority will be executed on first come first serve basis (Kunal Chandiramani, Rishabh Verma, 2019). Priority is defined externally or internally. Priorities which defined internally uses some measurable qualities or quantities to computer priority of a process (Chandra Shekar N, 2017). Priority algorithm arranges processes based on the priority on its queue. Other thing on priority scheduling, system gives priority on running state can be pre-emption (Ledina Hoxha Karteri, 2015). The last algorithm is First Come First Serve (FCFS). The process is executed based on which one arrives first. The one that exist in the queue will be executed the first time. The concept of FSCS is the same as FIFO (First In First Out) (Kunal Chandiramani, Rishabh Verma, 2019). FCFS is one simplest algorithm. Once the process is exist in the ready queue, it will be executed by the CPU until it is terminated. But, this algorithm has a high average waiting time (Hoger K. Omar, Kamal H. Jihad, n.d.).

In this journal, discussion will given about the variant of round robin scheduling algorithm thoroughly. For round robin algorithm, it is a known algorithm which usually used in real-time application. Context switching is one important concept in round robin scheduling algorithm. Round robin is a pre-emption algorithm. System gives allocation of a slice of context switching. If the context switching is finish, the current process will be preemptived, and will be put in the back of ready queue. The application of this is in real-time and embedded systems. But, the classic round robin algorithm has a high turn around time, small throughput, high waiting time, and high context switching (Hoger K. Omar, Kamal H. Jihad, n.d.) (Sakshi et al., 2022).

Our main subject for discussion in this journal is about priority round robin scheduling algorithm. In priority round robin scheduling algorithm, system provides each process an average share of time to utilize the CPU and gives a small responds time (Freire et al., 2021). Also the priority is given to each process. The higher the priority, the higher the chance the process will be executed. In this journal, the priority is given based on number. The higher the number, the less the priority. The idea of using priority is one best effective solution of this algorithm. The priority scheduling algorithm for round robin architecture is a modified version of simple round robin scheduling (Putra, 2020).

METHOD

This journal is consisting of five chapters. The first chapter is introduction. Here, we discuss the concept about the CPU process scheduling algorithm simulation and the basic concept of priority round robin scheduling algorithm. Existing process scheduling algorithms are also discussed. The second chapter is explanation about methodology of simulation. Discussions about OS-SIM is presented. The third chapter is about result of the simulation. Here, some displays of OS-SIM are discussed thoroughly. The fourth chapter is about discussion and analysis of the simulation result. The last chapter is conclusion and future works.

In this simulation the data of four processes scheduling are presented. Why four processes? This sampling technique is used because if it is too many than the explanation will be too long. Four processes are used so that it can be explained easily. The burst time of the four processes are chosen randomly. In real world, hundreds of processes will be running inside the system. So in this journal, only a small amount of processes will be analysed. That is the sampling technique that is used. Each process is given with priority. The methods in choosing the priority data is to show clearly the each processes will pre-empt the system by priority. The data priority, which are 1, 3, 2, 4 are chosen not arbitrary, but are chosen to show how the system pre-empt the running process. The quantum is 3. The quantum is chosen based on the scheduling system to show the analysis more deeply. Arrival times are 0, 1, 3, 4. The arrival time 0 means that the running process directly executed without waiting. Arrival time 1 means after 1 ms then the process arrives in the ready queue. The same happens with arrival times 3 and 4. These data are chosen to show at what time processes arrive in the ready queue.

Computer simulation is the process of mathematical modelling, done on a computer, that is designed to predict the behaviour of a physical system or the behaviour of a real-world. Computer simulations are a useful tool for mathematical modelling in many subject of computer systems (Wikipedia, 2022). Computer simulations can be used to gain insight into new technology and to imitate the performance of systems which is too complex to analyse. By simulations, computer can model situation in real-life, so that the working of the system can be seen and studied. In simulation, variable can be changed, and system can predict the behaviour of systems.

Computer simulation has become useful in modelling many scheduling algorithms in operating system. One example of using computer to simulate is by the use of OS-SIM. In such simulation, the model behaviour of process scheduling, memory management, disk, and file, can be implemented. Then, models are needed. Model that represent the characteristic and behaviour of the selected system or process. Computer is used to execute the simulation. OS-SIM is an application to simulate the characteristics or behaviours of system (Alexmazinho, 2022). Scheduling is best learned through implementation. OS-SIM, a simulator of process scheduling makes implementation easier.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v7i4.11665 p-ISSN : 2541-044X

OS-SIM is used to explore the behaviour of operating system. Each process can be analyse step by step to see when it accesses the CPU, where the system put the process in memory, and how long it accessed the system. OS-SIM is a modular system (Alexmazinho, 2022). It makes it easier to be understood. This will make implementation of different algorithms to be experimented easier. By this experiments, the user can get better understanding of the performance, behaviour, and characteristics of the operating system's elements.

In Figure 1. below, it can be seen the dashboard of OS-SIM Simulator. Five pull-down menus are displayed. Which are: File, Processes, Memory, File System, Disk, and Help. There are four major functions of operating systems to be simulated. Which are process scheduling, disk management, memory management, and file management. File Menu is for file manipulation. Menu Process is for process scheduling. Menu Memory is for memory management. Disk Menu is for disk saving simulation. The last menu, Help, is for information and help about this simulator.

FIGURE 1. SIMULATOR OS-SIM

RESULT

In this chapter discussion will be given on simulation in OS-SIM. There will be one example. The displays of OS-SIM will be given. There are four processes with arrival times, priorities, and burst time each:

TABLE 1. EXAMPLE OF PROCESS

Process	Arrival Time	Priority	Burst Time
A	0	1	4
B	1	3	8
C	3	2	7
D	4	4	6

Here there are four processes, Please take a look on table 1, The four processes, namely A, B, C, and D. Also given here, priority, arrival times and burst times. Priority 1 is the highest priority and priority 4 is the lowest priority. The total burst time is 25.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v7i4.11665 p-ISSN : 2541-044X

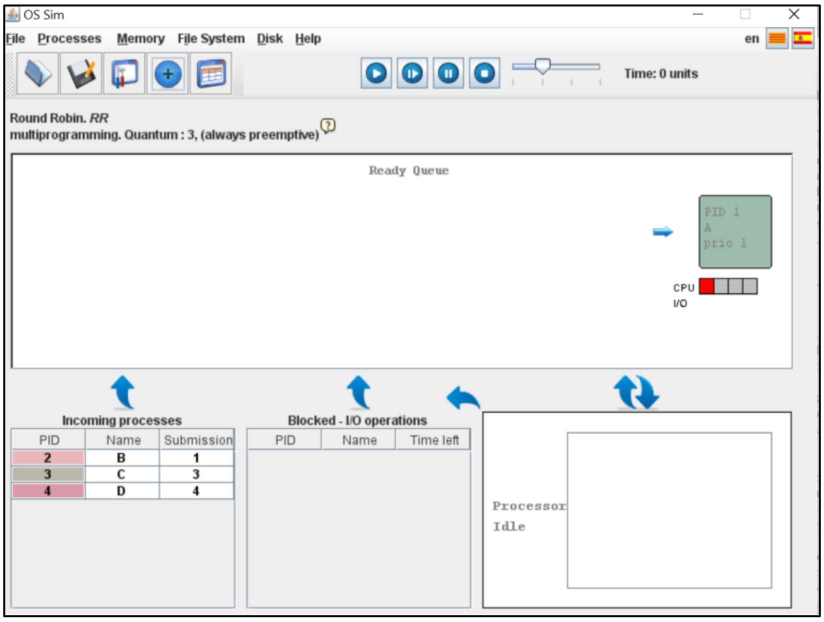

Figure 2. Step by Step Simulation at t=0

Please take a look on Figure 2. At t=0, process A gets in. Process A is 4 ms. B's arrival time is 1. C's arrival time is 3 and D's arrival time is 4. So at this moment, only A in the ready queue, since A's arrival time is 0. Processes B, C, D must wait four mili seconds, since this time is at 0.

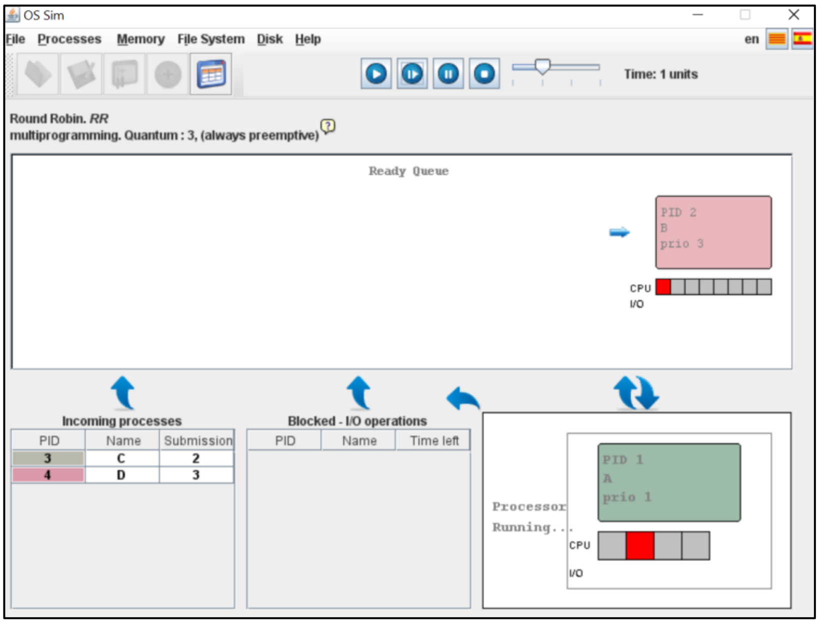

Figure 3. Step by Step Simulation at t=1

Please take a look on Figure 3. At t=1, B is placed in the ready queue. And at this time, A is executed for the first time. Since the priority of B is 3 and the priority of A is 1, which is higher then B. So, A is continue to be executed until quantum 3 is finished. So A is executed for 3 ms even though B's arrival time is at 1 ms. So that B is placed at this time, in the front of ready queue.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11665
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Figure 4. Step by Step Simulation at t=3

Please take a look on Figure 4. In this display, A is executed until its quantum is finished, namely until t=3. B is still in ready queue, waiting to be executed. The submission (arrival time) of C is 1 now, which means, in 1 ms, C will be placed in ready queue.

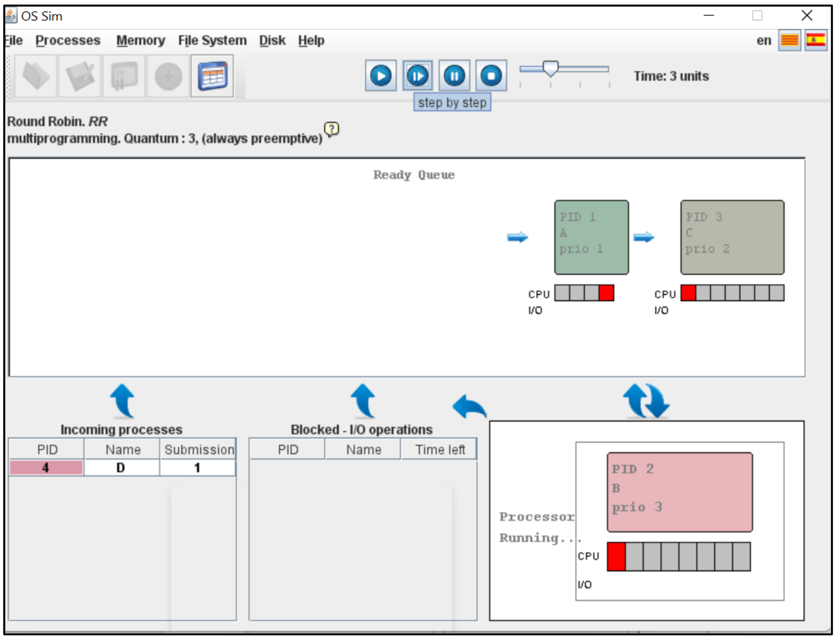

Figure 5. Step by Step Simulation at t=4

Please take a look on Figure 5. At t=4, B is executed, since B is in front of C in ready queue. C is placed behind B in the ready queue. For submission (arrival time) of D, it is left 1 ms. Which means in 1 ms, D will be placed in the ready queue.

*name of corresponding author

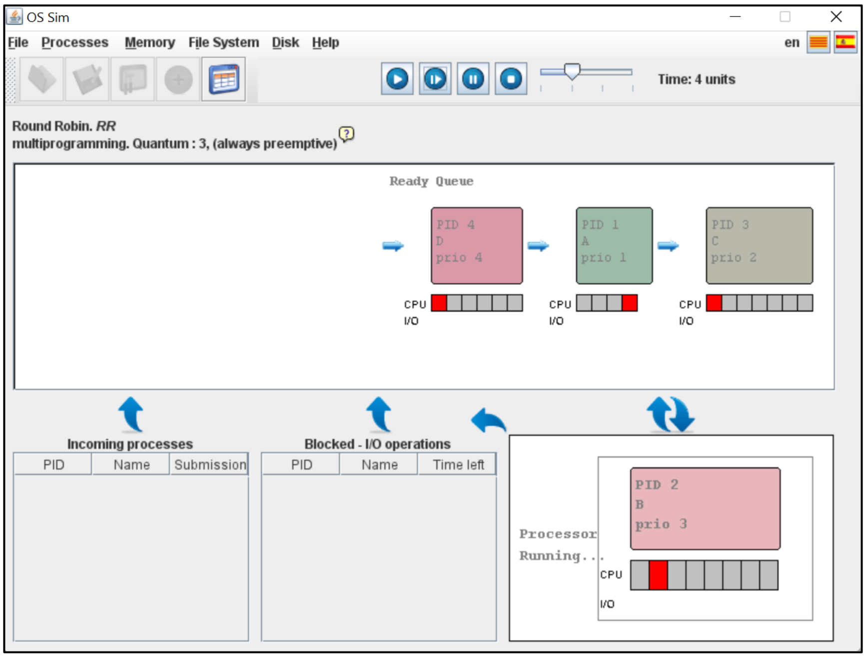

Figure 6. Step by Step Simulation at t=5

Please take a look on Figure 6. At t=5, B is still being executed. All processes, namely, A, B, C, and D, all of them are already placed in ready queue, waiting to be executed. In the front of ready queue is C. Followed by A and then D, in the back of the ready queue. In the mean time, B is still being executed.

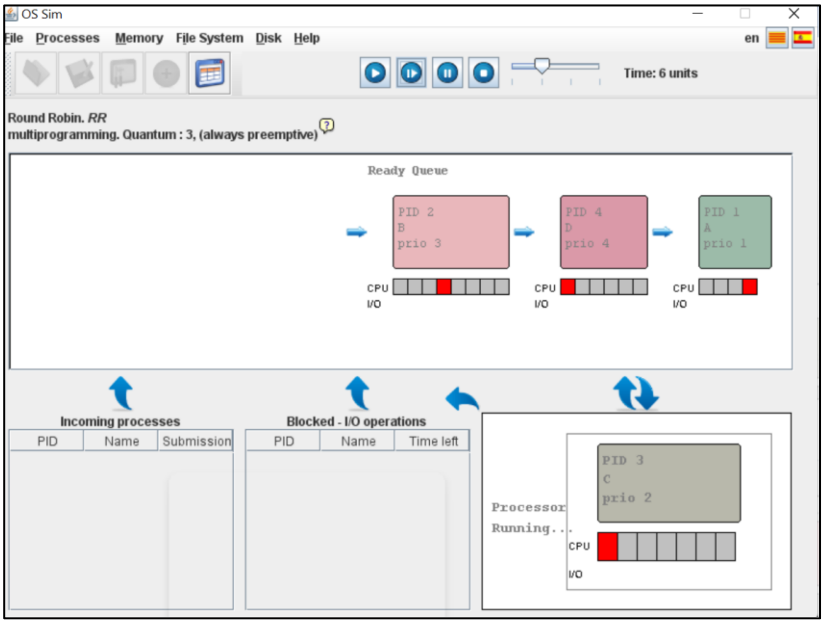

Figure 7. Step by Step Simulation at t =7

Please take a look on Figure 7. Process B is executed until 6 ms. At t=7, C is being executed. So in the front of ready queue is A now. Followed by D and then in the back is B. B is left five more ms. C is executed until t=9.

*name of corresponding author

 e-ISSN : 2541-2019
p-ISSN : 2541-044X

Figure 8. Step by Step Simulation at t=10

Please take a look on Figure 8. At t=10, process A is executed. Since A is in the front of ready queue. This is the last burst time of A. Since A has only 4 ms of burst time.

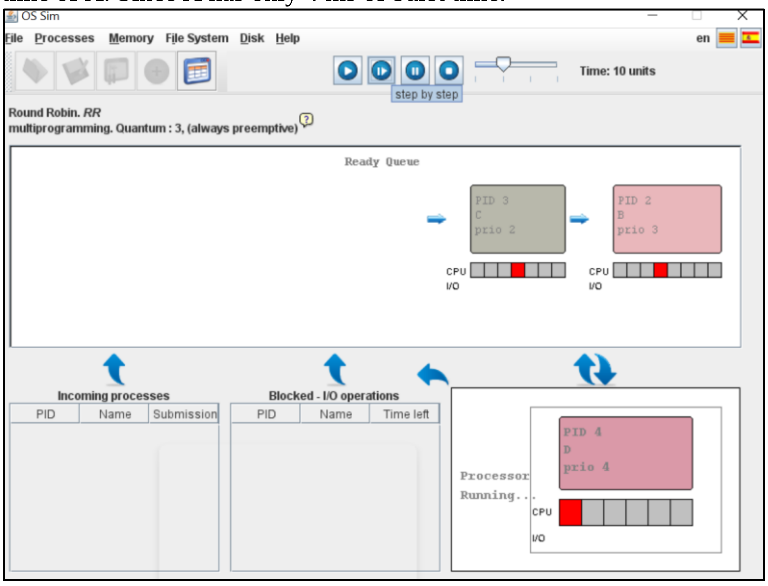

Figure 9. Step by Step Simulation at t=11

Please take a look on Figure 9. At t=11, D is executed. Since D is in the front of ready queue. D is executed as the quantum which is 3. D is executed until t=13.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v7i4.11665 p-ISSN : 2541-044X

Figure 10. Step by Step Simulation at t=14

Please take a look on Figure 10. At t=14, B is in the front of ready queue, so that, B is executed. B is executed as the quantum, which is 3. So, B is executed until t=16. Then, as in Figure 10, C is in the front of ready queue. Ready to be executed the next time.

Figure 11. Step by Step Simulation at t=17

Please take a look on Figure 11. At t=17, C is in the front of ready queue. Then, C is executed for 3 ms as the time quantum. Process C is executed until t=19. Behind C is D in the ready queue. Then in the back of the ready queue is B.

*name of corresponding author

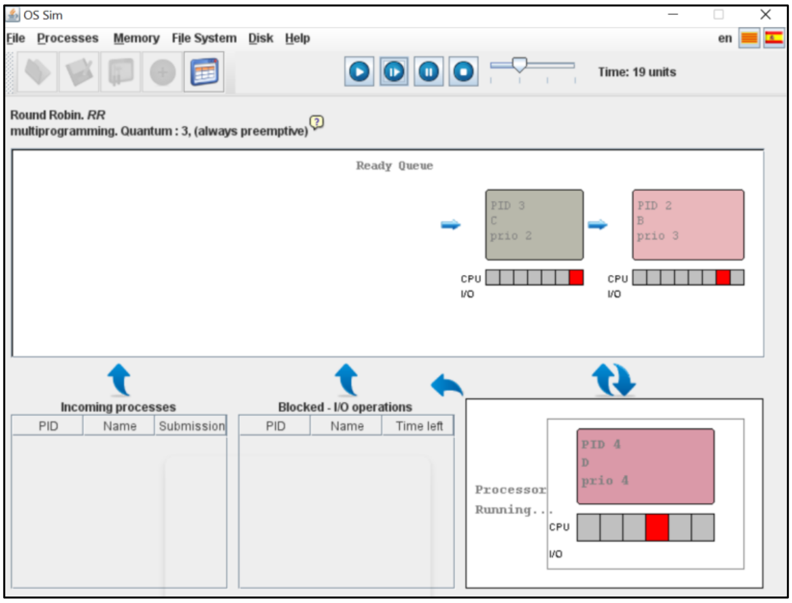

Figure 12. Step by Step Simulation at t=20

Please take a look on Figure 12. At t=20, D is executed. D is executed for 3 ms as the quantum. D is executed until t=22. Now in the ready queue is B. And in the back of the ready queue is C.

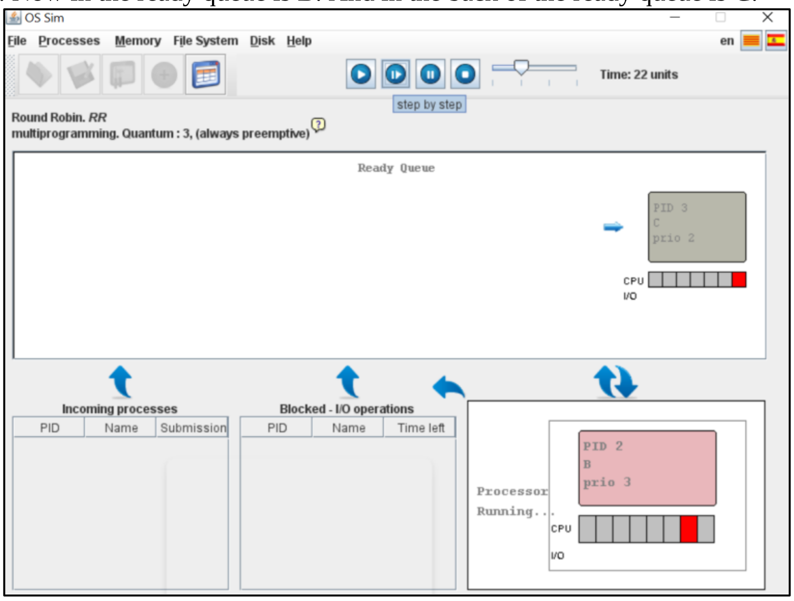

Figure 13. Step by Step Simulation at t=23

Please take a look on Figure 13. At t=23, B is executed. B only executed for 2 ms, since B is left 2 ms. B is executed until t=24. The last process, process C, now is in the front of ready queue.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11665
e-ISSN : 2541-2019
p-ISSN : 2541-044X

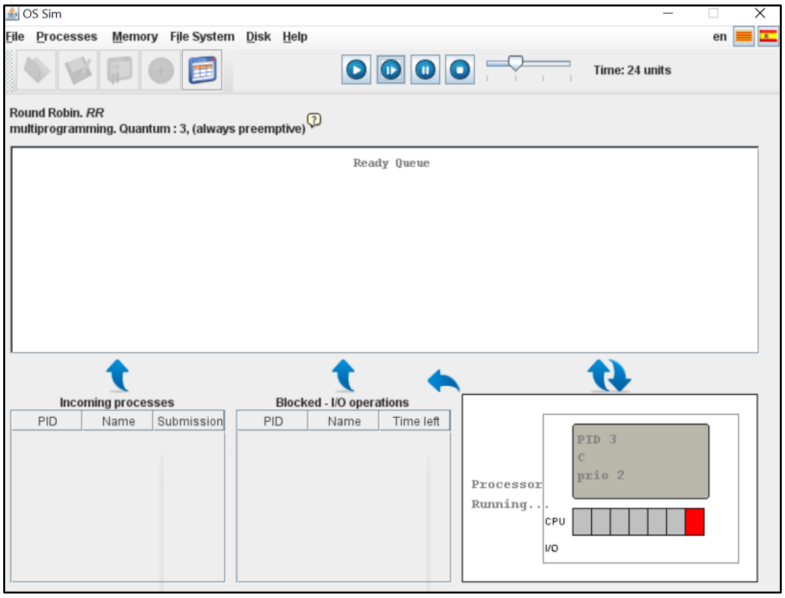

Figure 14. Step by Step Simultion at t=25

Please take a look on Figure 14. This is the last process. The last process is at t=25. C is executed. Since C is the last process in the ready queue. C is executed for only 1 ms.

DISCUSSION

Please take a look at gantt chart in Figure 15. A is executed for 3 ms. Process A is executed until finish, which is 3 ms. At t=1, B gets in to the front of ready queue. But, since the priority of A is higher then the priority of B, so, A continues to be executed until finish. The system compares the priority of process A and process B. A has 1 priority which is the highest. But, process B only has 3 priority, which is lower than process A. Then at t=4, B is executed until finished, at t=6. Then afterwards, C is executed from t=7 until t=9. Then, back A again to be executed, This is the last mili second of A. At t=11, D is executed for 3 ms, until t=13. Afterwards, B from t=14 is executed until t=16. At t=17, process C is executed until t=19. Then at t=20, the last process which has the lowest priority, which is 4, is executed for 3 ms. Until t=22. At t=23, Two last mili second of B is executed, until t=24. And at the last process in ready queue, C is executed for 1 ms only.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
A	A	A	B	B	B	C	C	C	A	D	D	D	B	B	B	C	C	C	D
21	22	23	24	25															
D	D	B	B	C															

Figure 15. Gantt Chart of the Processes

To get average waiting time, average turn around time, and average responds time. Please take a look on the Table 2. below. This is based on the calculation of the system, by the simulator automatically. The average responds time is 2.75 ms. The average waiting time is 12.00 ms. The average turn around time is 18.25 ms. For waiting time, it is for process A= 6, for process B and C, it is equal to 15 each. And for process D, it is 12. For turn around time, it is 10 for process A. It is 23 for process B. It is 22 for process B, and 18 for process D. For CPU usage utilization, for process A is 0.4. For process B is 0.3478. For process C is 0.3182. And for process D is 0.3333. There is no I/O utilization here, since it is not done here.

Table 2. Calculation of Turn Around Time, Waiting Time, and Responds Time

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11665
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Process Scheduling Information

Efficiency (%)	0.45
Throughput (processes/time unit)	0.07
Avg. Turnaround Time (time)	18.25
Avg. Waiting Time (time)	12.00
Avg. Response Time (time)	2.75

PID	Name	Priority	Submission	Periodic	CPU	Response	Waiting	Turnaround	% CPU	% IO
1	A	1	0	-	4	0	6	10	0.4	0.0
4	D	4	4	-	6	6	12	18	0.3333333...	0.0
2	B	3	1	-	8	2	15	23	0.3478260...	0.0
3	C	2	3	-	7	3	15	22	0.3181818...	0.0

CONCLUSION

In this journal, simulation of priority round robin scheduling algorithm is presented. With one example, by using four processes and using quantum=3. The average turn around time is 18.25 ms. And the Average Waiting Time is 12 ms. The Average Responds Time is 2.75 ms. OS-SIM is a well known operating system simulator. By using OS-SIM, it is easier to simulate the systems, and make it easy to understood the working process which is behind the scene. All processes in this example is finished at 25 ms. For future recommendations, other algorithm also can be simulated. Comparison between algorithm can also be done after simulation. It is also proposed to used other time quantum. Since in this journal, time quantum 3 is given. It is proposed for future works, other time quantum can be done also, like quantum=4 or quantum=5 and compare the results.

REFERENCES

Alexmazinho. (2022). *OS-SIM*. OS-Simulator. https://sourceforge.net/projects/oscsimulator/

Asma Joshita Trisha, S. B. (2019). A Combined Preemptive SJF and Preemptive Priority Algorithm to Enhance CPU Utilization. *International Journal of Computer Applications*, *177*(19), 26–30.

Chandra Shekar N, K. V. (2017). Analysis of Priority Scheduling Algorithm on the Basis of FCSF & SJF for Similar Priority Jobs. *International Journal of Engineering Research in Computer Science and Engineering*, *4*(3), 73–76.

Dhruv, R. (2019). Round Robin Scheduling Algorithm Based on Dynamic Time Quantum. *International Journal of Engineering and Advanced Technology (IJEAT)*, *X*(X), 593–595.

Freire, D. L., Frantz, R. Z., Roos-frantz, F., & Fernandes, V. B. (2021). *New developments in Round Robin algorithms and their applications : a systematic mapping study New developments in Round Robin algorithms and their applications : a systematic mapping study Daniela L . Freire , Rafael Z . Frantz , Fabricia Roos-Frantz Vit*. *January*.

Hoger K. Omar, Kamal H. Jihad, S. F. H. (n.d.). Comparative Analysis of the Essential CPU Scheduling Algorithms. *Bulletin of Electrical Engineering and Informatics*, *10*(5), 2742–2750.

Kunal Chandiramani, Rishabh Verma, S. M. (2019). A Modified Priority Preemptive Algorithm for CPU Scheduling. *International Confierence on Recent Trends in Advanced Computing 2019, ICRTAC 2019*, 363–369.

Ledina Hoxha Karteri, A. S. (2015). Preemptive and Non- Preemptive Priority Scheduling. *International Journal of Computer Science & Management Studies*, *19*(01), 1–5. www.ijcsms.com

m. LaxmiJeevani, T.S.P. Madhuri, Y. S. D. (2018). Improvised Round Robin Scheduling Algorithm and comparison with Existing Round Robin CPU Scheduling Algorithm. *IOSR Journal of Computer Engineering*, *20*(3), 1–4.

Putra, T. D. (2020). Analysis of Preemptive Shortest Job First (SJF) Algorithm in CPU Scheduling. *IJARCCE*, *9*(4), 41–45. https://doi.org/10.17148/ijarcce.2020.9408

Putra, T. D. (2022). Analysis of Priority Preemptive Scheduling Algorithm: Case Study. *Ijarcce*, *11*(1), 27–30. https://doi.org/10.17148/ijarcce.2022.11105

Putra, T. D., & Purnomo, R. (2021). Analisis Algoritma Round Robin pada Penjadwalan CPU. *Jurnal Ilmiah Teknologi Informasi Asia*, *15*(2), 85. https://doi.org/10.32815/jitika.v15i2.481

Putra, T. D., & Purnomo, R. (2022). Case Study : Improved Round Robin Algorithm. *Sinkron : Jurnal Dan Penelitian Teknik Informatika*, *7*(3), 950–956.

Sakshi, Sharma, C., Sharma, S., Kautish, S., A. M. Alsallami, S., Khalil, E. M., & Wagdy Mohamed, A. (2022). A new median-average round Robin scheduling algorithm: An optimal approach for reducing turnaround

*name of corresponding author

DOI : https://doi.org/10.33395/sinkron.v7i4.11665
e-ISSN : 2541-2019
p-ISSN : 2541-044X

and waiting time. *Alexandria Engineering Journal*, *61*(12), 10527–10538. https://doi.org/10.1016/j.aej.2022.04.006

Tri Dharma Putra, A. F. (2021). Comparison Between Simple Round Robin and Intelligent Round Robin Algorithms in CPU Scheduling. *International Journal of Advanced Research in Computer and Communication Engineering*, *10*(4), 86–90.

Vinay Kumar Reddy, E., & Aakash, K. (2021). A New Proposal of Smart Time Quantum for Round Robin Algorithm and Comparison with Existing Round Robin Variants. *International Research Journal of Engineering and Technology*, 1392–1402. www.irjet.net

Wikipedia. (2022). *Computer Simulation*. Wikipedia. https://en.wikipedia.org/wiki/Computer_simulation

*name of corresponding author