www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 2 Issue 12, Dec.2013 Page No. 3396-3399

SJRR CPU Scheduling Algorithm

Rakesh Patel , Mrs. Mili. Patel
Lecturer, Department of Information Technology ,Kirodimal Institute of Technology Raigarh

Lecturer, Department of Information Technology ,Kirodimal Institute of Technology Raigarh

Abstract :
The main objective of this paper is to introduce a new CPU algorithm called SJRR CPU Scheduling Algorithm which acts as preemptive based on the arrival time. The algorithm helps to improve the average waiting time of Round Robin algorithm in real time uni-processor-multi programming operating system. CPU Scheduling is the basis of multi-programmed operating system. The scheduler is responsible for multiplexing processes on the CPU. There are many scheduling algorithms available for a multi-programmed operating system like FCFS, SJF, Priority, Round Robin etc. The proposed algorithm is based on Round robin scheduling . In this paper, the results of the existing Round Robin algorithm is compared with the proposed algorithm.

Introduction:

Operating system performs variety of tasks in which scheduling is one of the basic task. Scheduling is heart of any computer system since it contains decision of giving resources between possible processes. Sharing of computer resources between multiple processes is also called scheduling . Process is a smallest work unit of a program which requires a set of resources for its execution that are allocated to it by the CPU. These processes are many in number and keep coming in a particular fashion, different scheduling techniques are employed that enable faster and efficient process execution thereby reducing the waiting time faced by each process and increasing CPU utilization. A process has five basic states namely NEW, Ready, Running, Waiting and Terminat .

Throughout its lifetime a process migrates between various scheduling queues by different schedulers until it gets terminated. These queues mainly contain the ready queue which contains set of processes ready for CPU response. The second queue is the device or the I/O queue which contains all the processes that are waiting for I/O response. The operating system must select processes for scheduling from these queues in a specific manner. This selection process using a particular scheduling technique is carried out by schedulers. Schedulers in general try to maximize the average performance of a system according to the given criterion. Scheduling algorithms are broadly classified into preemptive and non - preemptive scheduling disciplines.

The algorithm proposed in this article is preemptive in nature and attempts to give fair CPU execution time by focusing on average waiting time and turnaround time of a process.

Scheduling Parameters:

There are different scheduling algorithm with different characteristics which decide selection of process using different criteria for execution by CPU. The Criteria for a good scheduling algorithm depends, among others , on the following measures:-

A. ***CPU Utilization****:* It is the average fraction of time, during which the processor is busy .

B. ***Throughput:*** It refers to the amount of work completed in a unit of time. The number of processes the system can execute in a period of time. The higher the number, the more work is done by the system .

C. ***Waiting Time:*** The average period of time a process spends waiting. Waiting time may be expressed as turnaround time less the actual execution time .

D. ***Turnaround time****:* The interval from the time of submission of a process to the time of completion is the turnaround time.

E. ***Response time:*** Response time is the time from submission of a request until the first response is produced .

F. ***Priority****:* give preferential treatment to processes with higher priorities .

G. ***Fairness:*** Avoid the process from starvation. All the processes must be given equal opportunity to execute

.

EXISTING CPU SCHEDULING ALGORITHMS OVERVIEW:

A. FIRST COME FIRST SERVED (FCFS) Scheduling.

It is the simplest CPU Scheduling algorithm. The criteria of this algorithm is „the process that requests first, holds the CPU first" or which process enter the ready queue first is served first. The workload is processed in the order of arrival time, with no preemption . Once a process has been submitted to the CPU, it runs into completion without being interrupted. Such a technique is fair in the case of smaller processes but is quite unfair for long an unimportant job. Since FCFS does not involve context switching therefore it has minimal overhead. It has low throughput since long processes can keep processor occupied for a long time making small processes suffer. As a result waiting time, turnaround time and response time can be low .

B. Shortest Job First (SJF) Scheduling.

The criteria of this algorithm are which process having the smallest CPU burst, CPU is assigned to that process next. If two process having the same CPU burst time FCFS is used to break up the tie [3]. SJF can be worked as preemptive and non–preemptive in nature based on the arrival time and burst time of the processes. SJF reduces average waiting time of the processes as compared to FCFS. SJF favors shorter processes over longer ones which is an overhead as compared to FCFS. It selects the job with the smallest burst time ensuing CPU availability for other processes as soon as the current process reaches its completion. This prevents smaller processes from suffering behind larger processes in the ready queue for a longer time .

C. Priority Based Scheduling.

In this algorithm, priority is associated with each

process and on the basis of that priority CPU is allocated to the processes. Higher priority processes are executed first and lower priority processes are executed at the end. If multiple processes having the same priorities are ready to execute, control of CPU is assigned to these processes on the basis of FCFS [1]. Priority Scheduling can be preemptive and non-preemptive in nature.

D. Round Robin (RR) Scheduling.

It is a preemptive scheduling algorithm. It is designed especially for time sharing systems. In this algorithm, a small unit of time called time quantum or time slice is assigned to each process [2]. When the time quantum expired, the CPU is switched to another process. Performance of Round Robin totally depends on the size of the time quantum.

PROPOSED WORK: SJRR CPU SCHEDULING ALGORITHM :

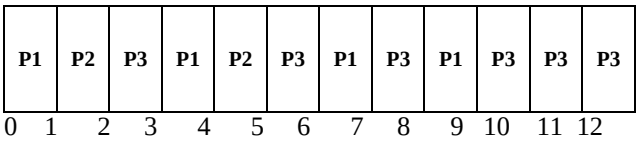
The proposed algorithm SJRR CPU Scheduling algorithm is preemptive in nature and based on Round robin scheduling mechanism . In this algorithm we will try to improve the average waiting time of Round robin scheduling by applying certain rules and regulation. In this research we try to show the comparison between traditional Round robin scheduling technique to proposed algorithm

For example suppose we have three different process with following properties

	Burst Time
P1	4
P2	2
P3	6

So here if we are using traditional Round Robin algorithm the scheduling is following

Suppose for above condition let the time quanta is of 1ns then the Gantt Chart for Round robin scheduling is



So the waiting time of each process is:-

P1 = 0 + 2 + 2 + 1 = 5

P2 = 1 + 2 = 3

P3 = 2 + 2 + 1 + 1 = 6

The average waiting time is = (5 + 3 + 6) / 3

= 4.66

In proposed algorithm the following steps are going to take:

- Select the sequence of process according to the to burst time. Means the process having minimum burst time should be select first then 2nd one. When we arrange the processes into increasing order to their burst time the average waiting time is going improve which is shown in bellow.
- The quanta should be equal to the shortest process which may affect to response time but due to this we can improve the average waiting time which is more important

In above example process P2 is the shortest process so we take the quanta according to the burst time of P2 i.e. 2 and then select the

sequence of process P2,p1 and then P3 according to proposed algorithm .

For above condition the time quanta is of 2ns then the Gantt Chart for proposed scheduling algorithm is

P2	P1	P3	P1	P3	P3

0 2 4 6 8 10 12

So the waiting time of each process is:-

P1 = 2 + 2 = 4

P2 = 0

P3 = 4 + 2 = 6

The average waiting time is = (4 + 0 + 6) / 3

= 3.33

Which is much better then traditional one. When we increase the number of process and the burst time of each process the performance of proposed algorithm
Going to increase as well.

Conclusion:

The paper presents a new CPU scheduling algorithm called SJRR CPU Scheduling Algorithm which is based on traditional Round Robin algorithm .Paper also contains The comparison of proposed algorithm with traditional Round Robin algorithm. The result of the comparison for different process sets using different scheduling algorithms .The Paper also explain how proposed algorithm improve the performance of traditional one

References:

[1] Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, "Operating System Concepts", Sixth Edition.

[2] Milan Milenkovic, "Operating Systems Concepts and Design", McGRAM-HILL, Computer Science Series, second edition.

[3] P. Balakrishna Prasad, "Operating Systems" Second Edition.

[4] A. Dhore "Opeating Systems", Technical Publications.

[5] M. Dietel, "Operating Systems", Pearson Education, Second Edition.

[6] http://en.wikipedia.org/wiki/Scheduling

[7] M Gary Nutt, "Operating systems – A Modern Perspective, Second Edition, Pearson Education, 2000.