

CSD 204 Lab

Lab 3: CPU Scheduling

[Dr. Sweta Kumari, Assistant Professor, SNioE]

Deadline - 16th Feb 2025, 11:59PM

Goal: This lab will improve your understanding of various non-preemptive & preemptive CPU scheduling such as FCFS, SJF, SRTF, Round-Robin, Priority, etc. (discussed in the class).

Before you begin

- Familiarize yourself with various CPU scheduling algorithms like FIRST COME FIRST SERVE (FCFS), SHORTEST JOB FIRST (SJF), SHORTEST REMAINING TIME FIRST (SRTF)/PREEMPTIVE SJF, ROUND-ROBIN, Priority.

Exercises

1. Write a c program to simulate the FCFS scheduling algorithm.

Hint: To calculate the average waiting time using the FCFS algorithm first the waiting time of the first process is kept zero and the waiting time of the second process is the burst time of the first process and the waiting time of the third process is the sum of the burst times of the first and the second process and so on. After calculating all the waiting times the average waiting time is calculated as the average of all the waiting times. FCFS mainly says first come first serve the algorithm which came first will be served first.

ALGORITHM:

Step 1: Start the process

Step 2: Accept the number of processes in the ready Queue

Step 3: For each process in the ready Q, assign the process name and the burst time

Step 4: Set the waiting of the first process as =0 and its burst time as its turnaround time

Step 5: for each process in the Ready Q calculate

a). $\text{Waiting time}(n) = \text{waiting time}(n-1) + \text{Burst time}(n-1)$

b). $\text{Turnaround time}(n) = \text{waiting time}(n) + \text{Burst time}(n)$

Step 6: Calculate

- a) Average waiting time = Total waiting Time / Number of process
- b) Average Turnaround time = Total Turnaround Time / Number of process

Step 7: Stop the process

2. Write a program in C to stimulate the CPU scheduling algorithm Shortest job first (Non-Preemption)

Hint: To calculate the average waiting time in the shortest job first algorithm the sorting of the process based on their burst time in ascending order then calculate the waiting time of each process as the sum of the bursting times of all the process previous or before to that process.

ALGORITHM:

Step 1: Start the process

Step 2: Accept the number of processes in the ready Queue

Step 3: For each process in the ready Q, assign the process id and accept the CPU burst time

Step 4: Start the Ready Q according the shortest Burst time by sorting according to lowest to highest burst time.

Step 5: Set the waiting time of the first process as $_0$ and its turnaround time as its burst time.

Step 6: Sort the processes names based on their Burt time

Step 7: For each process in the ready queue, calculate

a) Waiting time(n)= waiting time (n-1) + Burst time (n-1)

b) Turnaround time (n)= waiting time(n)+Burst time(n)

Step 8: Calculate

c) Average waiting time = Total waiting Time / Number of process

d)Average Turnaround time = Total Turnaround Time / Number of process

Step 9: Stop the process

3. Write a C program to simulate the Shortest Remaining Time First scheduling algorithm.
4. Write a C program to simulate the CPU scheduling algorithm round-robin.

Hint: To aim is to calculate the average waiting time. There will be a time slice, each process should be executed within that time-slice and if not it will go to the waiting state so first check whether the burst time is less than the time-slice. If it is less than it assign the waiting time to the sum of the total times. If it is greater than the burst-time then subtract the time slot from the actual burst time and increment it by time-slot and the loop continues until all the processes are completed.

ALGORITHM:

Step 1: Start the process

Step 2: Accept the number of processes in the ready Queue and time quantum (or) time slice

Step 3: For each process in the ready Q, assign the process id and accept the CPU burst time

Step 4: Calculate the no. of time slices for each process where No. of time slice for process (n) = burst time process (n)/time slice

Step 5: If the burst time is less than the time slice then the no. of time slices =1.

Step 6: Consider the ready queue is a circular Q, calculate

a) Waiting time for process (n) = waiting time of process(n-1)+ burst time of process(n-1)
+ the time difference in getting the CPU from process(n-1)

b) Turnaround time for process(n) = waiting time of process(n) + burst time of process(n)+
the time difference in getting CPU from process(n).

Step 7: Calculate

c) Average waiting time = Total waiting Time / Number of process

d) Average Turnaround time = Total Turnaround Time / Number of process

Step 8: Stop the process

5. Write a program in C to stimulate the CPU scheduling algorithm Priority scheduling (Non-Preemption)
6. Write a program in C to stimulate the CPU scheduling algorithm Priority scheduling (Preemption)

Submission Format:- You have to upload: (1) The source code in the following format: Assgn3Src-<Name>.c (2) Report: Assgn3Report-<Name>.pdf. Name the zipped document as: Assgn3-<Name>.zip.

Note: Please follow this naming convention mentioned above. make different files for codes for each question.

Grading Policy:- The policy for grading this assignment will be - (1) Design as described in the report and analysis of the results: 10% (2) Execution of the tasks based on the description in the readme: 80% (3) Code documentation and indentation: 10%.

Please note:

- All assignments for this course have a late submission policy of a penalty of 10% each day after the deadline of six days. After that, it will not be evaluated.

- All submissions are subject to plagiarism checks. Any case of plagiarism will be dealt with severely.