Lab: CPU Scheduling

Part A: FIRST COME FIRST SERVE

AIM: To write a c program to simulate the CPU scheduling algorithm First Come First Serve (FCFS)

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). Waiting time (n) = waiting time (n-1) + Burst time (n-1)
        b). Turnaround time (n)= waiting time(n)+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
```

Part B. SHORTEST JOB FIRST:

AIM: To 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
```

Part C: ROUND ROBIN:

AIM: To write a program in C 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 fromprocess(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 ofprocess
Step 8: Stop the process
```

Part D: PRIORITY SCHEDULING:

AIM: To write a c program to simulate the CPU scheduling priorityalgorithm.

Hint: To calculate the average waiting time in the priority algorithm, sort the burst times according to their priorities and then calculate the average waiting time of the processes. The waiting time of each process is obtained by summing up the burst times of all the previous processes.

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: Sort the ready queue according to the priority number.
Step 5: Set the waiting of the first process as _0‘ and its burst time as its turnaround time
Step 6: Arrange the processes based on process priority
Step 7: For each process in the Ready Q calculate
Step 8: for each process in the Ready Q
        calculate
        a) Waiting time(n)= waiting time (n-1) + Burst time (n-1)
        b) Turnaround time (n)= waiting time(n)+Burst time(n)
Step 9: Calculate
        c) Average waiting time = Total waiting Time / Number of process
        d) Average Turnaround time = Total Turnaround Time / Number of process Print the results
        in an order.
Step10: Stop
```

BONUS QUESTION

Part E: Preemptive SJF:

AIM: To write a program in C to simulate the Shortest Remaining Time First scheduling algorithm.