



Mawlana Bhashani Science And Technology University

Lab-Report

Report No : 07

Course Code : ICT-3110

Course Title : Operating System Lab.

Date of Performance :

Date of Submission : 20/09/2020

Submitted By:

Name: Md Abdur Rahim

ID: IT-18024

3rd Year 1st Semester

Session : 2017-18

Dept. of ICT

MBSTU

Submitted To:

Nazrul Islam

Assistant Professor

Dept. of ICT

MBSTU

Experiment No : 07

Experiment Name: Implementation of FCFS Scheduling Algorithm.

FCFS Scheduling Algorithm :

The First Come First Served (FCFS) Scheduling Algorithm is the simplest one. In this algorithm the set of ready processes is managed as FIFO (first in-first-out) Queue. The processes are serviced by the CPU until completion in order of their entering in the FIFO queue.

A process once allocated the CPU keeps it until releasing the CPU either by terminating or requesting I/O.

For example, interrupted process is allowed to run after interrupt handling is done with.

Target: To write the c program to implement scheduling algorithm for first come first serve scheduling (FCFS).

Description:

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

Code:

```
#include<iostream>

using namespace std;

void findWaitingTime(int processes[], int n, int bt[], int wt[])
{
    wt[0] = 0;
    for (int i = 1; i < n ; i++)
    {
        wt[i] =  bt[i-1] + wt[i-1];
    }
}

void findTurnAroundTime( int processes[], int n, int bt[], int wt[], int tat[])
{
    for (int i = 0; i < n ; i++)
    {
        tat[i] = bt[i] + wt[i];
    }
}
```

```

void findAverageTime( int processes[], int n, int bt[])
{
    int wt[n], tat[n], total_wt = 0, total_tat = 0;

    findWaitingTime(processes, n, bt, wt);

    findTurnAroundTime(processes, n, bt, wt, tat);

    cout << "Processes  " << " Burst time  " << " Waiting time  " << " Turn
around time\n";

    for (int i = 0; i < n; i++)
    {
        total_wt = total_wt + wt[i];

        total_tat = total_tat + tat[i];

        cout << "      " << i+1 << "\t\t" << bt[i] << "\t      " << wt[i] << "\t\t  "
<< tat[i] << endl;
    }

    cout << "Average waiting time = " << (float)total_wt / (float)n;

    cout << "\nAverage turn around time = " << (float)total_tat / (float)n;
}

int main()
{
    int processes[] = { 1, 2, 3, 4};

    int n = sizeof processes / sizeof processes[0];

    int burst_time[] = {21, 3, 6, 2};

    findAverageTime(processes, n, burst_time);

    return 0;
}

```

Output:

Processes	Burst time	Waiting time	Turn around time
1	21	0	21
2	3	21	24
3	6	24	30
4	2	30	32

Average waiting time = 18.75
Average turn around time = 26.75

Discussion:

This lab help us to learn about FCFS scheduling algorithm. we have implemented this algorithm using c language. Using this algorithm we can find waiting and turnaround time.