



Mawlana Bhashani Science and Technology University

Lab -Report

Report No:07

Course code: ICT-3110

Course title: Operating System Lab

Date of Performance:26-08-2020

Date of Submission:02-09-2020

Submitted by

Name: Ruku Shikder
ID: IT-18057
3th year 1st semester
Session: 2017-2018
Dept. of ICT
MBSTU.

Submitted To

Nazrul Islam
Assistant Professor
Dept. of ICT
MBSTU.

Experiment No: 07

Experiment Name: Implementation of FCFS Scheduling Algorithm

- What is FCFS Scheduling Algorithm? [SEP]
- How to implementation in C? [SEP]

FCFS Scheduling Algorithm:

Given n processes with their burst times and arrival times, the task is to find average waiting time and average turn around time using FCFS scheduling algorithm.

FIFO simply queues processes in the order they arrive in the ready queue. Here, the process that comes first will be executed first and next process will start only after the previous gets fully executed.

- Completion Time: Time at which process completes its execution.
- Turn Around Time: Time Difference between completion time and arrival time. Turn Around Time = Completion Time – Arrival Time
- Waiting Time(W.T): Time Difference between turn around time and burst time. Waiting Time = Turn Around Time – Burst Time

Service Time : Service time means amount of time after which a process can start execution. It is summation of burst time of 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 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

Implementation in C:

Code:

```
#include<stdio.h>

int main()
{
    int n,bt[20],wt[20],tat[20],avwt=0,avtat=0,i,j;
    printf("Enter total number of processes(maximum 20):");
    scanf("%d",&n);

    printf("\nEnter Process Burst Time\n");
    for(i=0;i<n;i++)
    {
        printf("P[%d]:",i+1);
        scanf("%d",&bt[i]);
    }

    wt[0]=0; //waiting time for first process is 0

    //calculating waiting time
    for(i=1;i<n;i++)
    {
        wt[i]=0;
        for(j=0;j<i;j++)
            wt[i]+=bt[j];
    }

    printf("\nProcess\tBurst Time\tWaiting Time\tTurnaround Time");

    //calculating turnaround time
    for(i=0;i<n;i++)
    {
        tat[i]=bt[i]+wt[i];
        avwt+=wt[i];
        avtat+=tat[i];
        printf("\nP[%d]\t\t%d\t\t%d\t\t%d",i+1,bt[i],wt[i],tat[i]);
    }

    avwt/=i;
    avtat/=i;
    printf("\n\nAverage Waiting Time:%d",avwt);
    printf("\n\nAverage Turnaround Time:%d",avtat);

    return 0;
}
```

Output:

```
ruku@hp-envy-notebook: ~/Desktop
ruku@hp-envy-notebook:~$ cd Desktop
ruku@hp-envy-notebook:~/Desktop$ gcc fcfs.c -o fcfs
ruku@hp-envy-notebook:~/Desktop$ ./fcfs
Enter total number of processes(maximum 20):4

Enter Process Burst Time
P[1]:20
P[2]:4
P[3]:5
P[4]:5

Process          Burst Time    Waiting Time   Turnaround Time
P[1]              20            0              20
P[2]              4            20            24
P[3]              5            24            29
P[4]              5            29            34

Average Waiting Time:18
Average Turnaround Time:26ruku@hp-envy-notebook:~/Desktop$
```

Discussion:

From this lab experiment I learn FCFS Scheduling Algorithm implementation using linux operating system. Completion Time, Turn Around Time, Around Time, Waiting Time(W.T), Waiting Time is help me to implement other scheduling algorithm.