



جامعة المستقبل  
AL MUSTAQBAL UNIVERSITY

كلية العلوم

قسم الأنظمة الطبية الذكية

Intelligent Medical Systems Department

Lab (2)

# CPU scheduling algorithms

**Subject: OPERATING SYSTEMS**

**Class: Second**

**Lecturer: Dr.Ahmad almhanna**

Eng .Jumana altahier

Programar :Ahyab hashim



**Write a C program simulate the following CPU scheduling algorithms:**

## a)FCFS

### DESCRIPTION

Assume all the processes arrive at the same time.

#### FCFS CPU SCHEDULING ALGORITHM

For FCFS scheduling algorithm, read the number of processes/jobs in the system, their CPU burst times. The scheduling is performed on the basis of arrival time of the processes irrespective of their other parameters. Each process will be executed according to its arrival time. Calculate the waiting time and turnaround time of each of the processes accordingly.

### CPU SCHEDULING

Maximum CPU utilization obtained with multiprogramming

CPU–I/O Burst Cycle – Process execution consists of a cycle of CPU execution and I/O wait

CPU burst distribution

### First-Come, First-Served (FCFS) Scheduling

Process	Burst Time
P1	24
P2	3
P3	3

Suppose that the processes arrive in the order: P1 , P2 , P3

The Gantt Chart for the schedule is:



Waiting time for P1 = 0; P2 = 24; P3 = 27

Average waiting time:  $(0 + 24 + 27)/3 = 17$



## ALGORITHM

1. Start
2. Declare the array size
3. Read the number of processes to be inserted
4. Read the Burst times of processes
5. calculate the waiting time of each process  $wt[i+1]=bt[i]+wt[i]$
6. calculate the turnaround time of each process  $tt[i+1]=tt[i]+bt[i+1]$
7. Calculate the average waiting time and average turnaround time.
8. Display the values
9. Stop



## PROGRAM:

```
#include<stdio.h>
void main()
{
    int i,j,bt[10],n,wt[10],tt[10],w1=0,t1=0;
    float aw,at;
    printf("enter no. of processes:\n");
    scanf("%d",&n);
    printf("enter the burst time of processes:");
    for(i=0;i<n;i++)
        scanf("%d",&bt[i]);
    for(i=0;i<n;i++)
    {
        wt[0]=0;
        tt[0]=bt[0];
        wt[i+1]=bt[i]+wt[i];
        tt[i+1]=tt[i]+bt[i+1];
        w1=w1+wt[i];
        t1=t1+tt[i];
    }
    aw=w1/n;
    at=t1/n;
    printf("\nbt\t wt\t tt\n");
    for(i=0;i<n;i++)
        printf("%d\t %d\t %d\n",bt[i],wt[i],tt[i]);
    printf("aw=%f\n,at=%f\n",aw,at);
}
```



INPUT

Enter no of processes

3

enter bursttime

12

8

20

EXPECTED OUTPUT

bt wt tt

12 0 12

8 12 20

20 20 40

aw=10.666670

at=24.00000



Intelligent Medical Systems Department

Operating system– Lab(2)  
Second stage

---

