|  |  |
| --- | --- |
| **Ex.No:5.c** | **CPU SCHEDULING ALGORITHMS** |
| **FCFS** |

# AIM:

To write a C program for implementation of FCFS and SJF scheduling algorithms.

# ALGORITHM:

Step 1: Inside the structure declare the variables.

Step 2: Declare the variable i,j as integer,totwtime and totttime is equal to zero. Step 3: Get the value of „n‟ assign pid as I and get the value of p[i].btime.

Step 4: Assign p[0] wtime as zero and tot time as btime and inside the loop calculate wait time and turnaround time.

Step 5: Calculate total wait time and total turnaround time by dividing by total number of process.

Step 6: Print total wait time and total turnaround time. Step 7: Stop the program.

# PROGRAM:

#include<stdio.h> #include<stdlib.h> struct fcfs

{

int pid; int btime; int wtime; int ttime;

} p[10];

int main()

{

int i,n;

int towtwtime=0,totttime=0; printf("\n fcfs scheduling...\n"); printf("enter the no of process"); scanf("%d",&n); for(i=0;i<n;i++)

{

p[i].pid=1;

printf("\n burst time of the process”); scanf("%d",&p[i].btime);

 }

p[0].wtime=0; p[0].ttime=p[0].btime; totttime+=p[i].ttime; for(i=0;i<n;i++)

{

p[i].wtime=p[i-1].wtime+p[i-1].btim

p[i].ttime=p[i].wtime+p[i].btime; totttime+=p[i].ttime; towtwtime+=p[i].wtime;

}

for(i=0;i<n;i++)

{{

printf("\n waiting time for process”); printf("\n turn around time for process”); printf("\n");

}}

printf("\n total waiting time :%d", totwtime ); printf("\n average waiting time :%f",(float)totwtime/n); printf("\n total turn around time :%d",totttime);

printf("\n average turn around time: :%f",(float)totttime/n);

}