first come, first served (FCFS )ALGORITHM

AIM:

PROGRAM TO IMPLEMENT first come, first served using c language.

First Come, First Served (FCFS) also known as First In, First Out(FIFO) is the CPU scheduling algorithm in which the CPU is allocated to the processes in the order they are queued in the ready queue.

FCFS follows non-preemptive scheduling which mean once the CPU is allocated to a process it does not leave the CPU until the process will not get terminated or may get halted due to some I/O interrupt.

ALGORITHM:

1- Input the processes along with their burst time (bt).

2- Find waiting time (wt) for all processes.

3- As first process that comes need not to wait so

waiting time for process 1 will be 0 i.e. wt[0] = 0.

4- Find **waiting time** for all other processes i.e. for

process i ->

wt[i] = bt[i-1] + wt[i-1] .

5- Find **turnaround time** = waiting\_time + burst\_time

for all processes.

6- Find **average waiting time** =

total\_waiting\_time / no\_of\_processes.

7- Similarly, find **average turnaround time** =

total\_turn\_around\_time / no\_of\_processes.

OR

ALGORITHM:

Start

Step 1-> In function int waitingtime(int proc[], int n, int burst\_time[], int wait\_time[])

   Set wait\_time[0] = 0

   Loop For i = 1 and i < n and i++

      Set wait\_time[i] = burst\_time[i-1] + wait\_time[i-1]

   End For

Step 2-> In function int turnaroundtime( int proc[], int n, int burst\_time[], int wait\_time[], int tat[])

   Loop For  i = 0 and i < n and i++

      Set tat[i] = burst\_time[i] + wait\_time[i]

   End For

Step 3-> In function int avgtime( int proc[], int n, int burst\_time[])

   Declare and initialize wait\_time[n], tat[n], total\_wt = 0, total\_tat = 0;

   Call waitingtime(proc, n, burst\_time, wait\_time)

   Call turnaroundtime(proc, n, burst\_time, wait\_time, tat)

   Loop For  i=0 and i<n and i++

      Set total\_wt = total\_wt + wait\_time[i]

      Set total\_tat = total\_tat + tat[i]

      Print process number, burstime wait time and turnaround time

   End For

   Print "Average waiting time =i.e. total\_wt / n

   Print "Average turn around time = i.e. total\_tat / n

Step 4-> In int main()

   Declare the input int proc[] = { 1, 2, 3}

   Declare and initialize n = sizeof proc / sizeof proc[0]

   Declare and initialize burst\_time[] = {10, 5, 8}

   Call avgtime(proc, n, burst\_time)

Stop

Table

Description automatically generated

PROGRAM:

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated with medium confidence

CODE:

#include<stdio.h>

int main()

{

int n,sum=0,bt[10]={0},tat[10]={0},wt[10]={0},at[10]={0},ct[10]={0};

float totalTAT=0,totalWT=0;

scanf("%d",&n);

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

scanf("%d",&bt[i]);

}

for(int j=0;j<n;j++){

sum+=bt[j];

ct[j]+=sum;

}

for(int k=0;k<n;k++){

tat[k]=ct[k]-at[k];

totalTAT+=tat[k];

}

for(int k=0;k<n;k++){

wt[k]=tat[k]-bt[k];

totalWT+=wt[k];

}

printf("%f\n",totalWT/n);

printf("%f\n",totalTAT/n);

return 0;

}

RESULT:

Table

Description automatically generated

RESULT:

FCFS PROGRAM WAS IMPLEMENT SUCCESSFULLY USING C LANGUAGE.