Experiment: 4

**Aim:** Write a program to implement FCFS (First Come First Serve) algorithm with same arrival time.

It is non-preemptive algorithm which means the process priority does not matter. If a process with very least priority is being executed and allow a sudden some high priority process arises like avoid to interrupt the system crash.

Then it will have to wait and hence system is crash just because of improper scheduling.

Program:

#include<stdio.h>

int main()

{

printf(">FCFS ( First Come First Serve ) algorithm with same arrival time.\n");

int i,sum=0;

int num;

int arvTime[15],bTime[15];

int serTime[15],wtTime[15];

float AvgWtTime,AvgTaTime;

printf("\nEnter number of process : ");

scanf("%d",&num); printf("\n");

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

{

arvTime[i]=0;

}

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

{

printf("Enter burst time of P%d : ",i+1);

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

}

serTime[0]=0;

for(i=1;i<num;i++){

serTime[i]=serTime[i-1]+bTime[i-1];

}

for(i=0;i<num;i++){

wtTime[i]=serTime[i]-arvTime[i];

}

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

{

sum=sum+serTime[i];

}

AvgTaTime = (float)sum/num;

sum=0;

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

{

sum=sum+wtTime[i];

}

AvgWtTime = (float)sum/num;

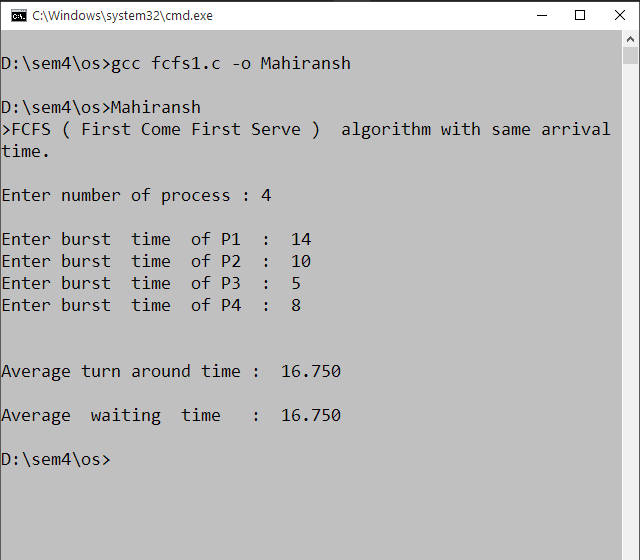
printf("\n\nAverage turn around time : %.3f",AvgTaTime);

printf("\n\nAverage waiting time : %.3f",AvgWtTime);

printf("\n");

}

Output:



Experiment: 5

**Aim:** Write a program to implement FCFS (First Come First Serve) algorithm with different arrival time.

It is same as before experiment but only arrival times of processes are different non-preemptive algorithm which means the process priority does not matter. If a process with very least priority is being executed and allow a sudden some high priority process arises like avoid to interrupt the system crash.

Then it will have to wait and hence system is crash just because of improper scheduling.

Program:

#include<stdio.h>

int main()

{

printf("FCFS(First Come First Serve) algorithm with different arrival time.\n");

int i,sum=0;

int num;

int arvTime[15],bTime[15];

int serTime[15],wtTime[15];

float AvgWtTime,AvgTaTime;

printf("\nEnter number of process : ");

scanf("%d",&num);

printf("\n");

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

{

printf("Enter arrival time of P%d : ",i+1);

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

printf("Enter burst time of P%d : ",i+1);

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

printf("\n");

}

serTime[0]=0;

for(i=1;i<num;i++){

serTime[i]=serTime[i-1]+bTime[i-1];

}

for(i=0;i<num;i++){

wtTime[i]=serTime[i]-arvTime[i];

}

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

{

sum=sum+serTime[i];

}

AvgTaTime = (float)sum/num;

sum=0;

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

{

sum=sum+wtTime[i];

}

AvgWtTime = (float)sum/num;

printf("\n\nAverage turn around time : %.3f",AvgTaTime);

printf("\n\nAverage waiting time : %.3f",AvgWtTime);

printf("\n");

}

Output:

