**ASSIGNMENT 10**

**TAMOJIT DAS A/7/C1**

Priority Preemptive

#include<stdio.h>

struct Job{

int Pr,AT,BT,CT,TAT,WT;

};

typedef struct Job Job;

Job Job\_Array[10];

float get\_avg\_TAT(int N){

float f=0;

int i=0;

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

f=f+Job\_Array[i].TAT;

}

return f/(N);

}

float get\_avg\_WT(int N){

float f=0;

int i=0;

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

f=f+Job\_Array[i].WT;

}

return f/(N);

}

int get\_max\_priority(int Time,int N){

int max=0;

int i=0,Index=0,flag=0;

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

if(Job\_Array[i].AT<=Time && Job\_Array[i].BT>0 && Job\_Array[i].Pr>max){

max=Job\_Array[i].Pr;

Index=i;

flag=1;

}

}

if(flag==0){

return -1;

}

return Index;

}

void main(){

int N=5,Timer=0,count=0,tmp\_index=0;

float Avg\_TAT,Avg\_WT;

Job\_Array[0]=(Job){4,0,5};

Job\_Array[1]=(Job){3,1,4};

Job\_Array[2]=(Job){6,1,3};

Job\_Array[3]=(Job){8,2,6};

Job\_Array[4]=(Job){5,3,8};

int Initial\_BT[]={5,4,3,6,8};

while(count<N){

tmp\_index=get\_max\_priority(Timer,N);

if(tmp\_index!=-1){

Job\_Array[tmp\_index].BT=Job\_Array[tmp\_index].BT-1;

Timer++;

if(Job\_Array[tmp\_index].BT==0){

count++;

Job\_Array[tmp\_index].CT=Timer;

Job\_Array[tmp\_index].TAT=Job\_Array[tmp\_index].CT-Job\_Array[tmp\_index].AT;

Job\_Array[tmp\_index].WT=Job\_Array[tmp\_index].TAT-Initial\_BT[tmp\_index];

}

}

}

printf("\n%s \t%s \t%s \t%s \t%s","AT","BT","CT","TAT","WT");

for(count=0;count<N;count++){

printf("\n%d \t%d \t%d \t%d \t%d",Job\_Array[count].AT,Initial\_BT[count],Job\_Array[count].CT,Job\_Array[count].TAT,Job\_Array[count].WT);

}

Avg\_TAT=get\_avg\_TAT(N);

Avg\_WT=get\_avg\_WT(N);

printf("\n\nAvg TAT: %f \tAvg WT: %f",Avg\_TAT,Avg\_WT);

}

