**OPERATING SYSTEMS LAB**

**AISHWARYA GS**

**CB.EN.U4CYS21004**

SHORTEST JOB FIRST

#include<stdio.h>

int main()

{

int bt[20],p[20],wt[20],tat[20],i,j,n,total=0,pos,temp;

float avg\_wt,avg\_tat;

printf("Enter number of process\n");

scanf("%d",&n);

printf("Enter Burst Time\n");

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

{

printf("p%d:",i+1);

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

p[i]=i+1;

}

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

{

pos=i;

for(j=i+1;j<n;j++)

{

if(bt[j]<bt[pos])

pos=j;

}

temp=bt[i];

bt[i]=bt[pos];

bt[pos]=temp;

temp=p[i];

p[i]=p[pos];

p[pos]=temp;

}

wt[0]=0;

for(i=1;i<n;i++)

{

wt[i]=0;

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

wt[i]+=bt[j];

total+=wt[i];

}

avg\_wt=(float)total/n;

total=0;

printf("Process\t, Burst Time\t, Waiting Time\t, Turnaround Time\t");

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

{

tat[i]=bt[i]+wt[i];

total+=tat[i];

printf("%p\t, %d\t, %d\t, %d\t",p[i],bt[i],wt[i],tat[i]);

}

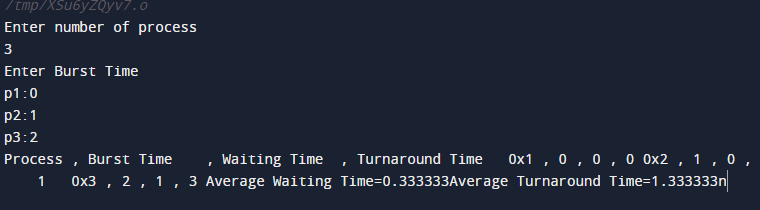
avg\_tat=(float)total/n;

printf("Average Waiting Time=%f",avg\_wt);

printf("Average Turnaround Time=%fn",avg\_tat);

}

OUTPUT



FIRST COME FIRST SERVE

#include <stdio.h>

int waitingtime(int proc[], int n,

int burst\_time[], int wait\_time[]) {

wait\_time[0] = 0;

for (int i = 1; i < n ; i++ )

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

return 0;

}

int turnaroundtime( int proc[], int n,

int burst\_time[], int wait\_time[], int tat[]) {

int i;

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

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

return 0;

}

int avgtime( int proc[], int n, int burst\_time[]) {

int wait\_time[n], tat[n], total\_wt = 0, total\_tat = 0;

int i;

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

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

printf("Processes Burst Waiting Turn around");

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

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

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

printf(" %d\t %d\t\t %d \t%d", i+1, burst\_time[i], wait\_time[i], tat[i]);

}

printf("Average waiting time = %f", (float)total\_wt / (float)n);

printf("Average turn around time = %f", (float)total\_tat / (float)n);

return 0;

}

int main() {

int proc[] = { 1, 2, 3};

int n = sizeof proc / sizeof proc[0];

//Burst time of all processes

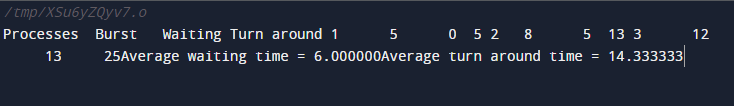
int burst\_time[] = {5, 8, 12};

avgtime(proc, n, burst\_time);

return 0;

}

OUTPUT



ROUND ROBIN

#include<stdio.h>

#include<conio.h>

void main()

{

int i, NOP, sum=0,count=0, y, quant, wt=0, tat=0, at[10], bt[10], temp[10];

float avg\_wt, avg\_tat;

printf(" Total number of process in the system: ");

scanf("%d", &NOP);

y = NOP;

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

{

printf("\n Enter the Arrival and Burst time of the Process[%d]\n", i+1);

printf(" Arrival time is: \t");

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

printf(" \nBurst time is: \t");

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

temp[i] = bt[i];

}

printf("Enter the Time Quantum for the process: \t");

scanf("%d", &quant);

printf("\n Process No \t\t Burst Time \t\t TAT \t\t Waiting Time ");

for(sum=0, i = 0; y!=0; )

{

if(temp[i] <= quant && temp[i] > 0)

{

sum = sum + temp[i];

temp[i] = 0;

count=1;

}

else if(temp[i] > 0)

{

temp[i] = temp[i] - quant;

sum = sum + quant;

}

if(temp[i]==0 && count==1)

{

y--;

printf("\nProcess No[%d] \t\t %d\t\t\t\t %d\t\t\t %d", i+1, bt[i], sum-at[i], sum-at[i]-bt[i]);

wt = wt+sum-at[i]-bt[i];

tat = tat+sum-at[i];

count =0;

}

if(i==NOP-1)

{

i=0;

}

else if(at[i+1]<=sum)

{

i++;

}

else

{

i=0;

}

}

avg\_wt = wt \* 1.0/NOP;

avg\_tat = tat \* 1.0/NOP;

printf("\n Average Turn Around Time: \t%f", avg\_wt);

printf("\n Average Waiting Time: \t%f", avg\_tat);

getch();

}

OUTPUT

