**PRACTICAL 8**

Priyanka Bhatt

B.tech CSE 5th Sem

2294053

**Date:** 24-09-24

**OBJECTIVE**

**Program to Implement Round Robin scheduling algorithm**

**CODE:**

#include<stdio.h>

int 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: ");

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: ");

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

printf(" Burst time is: ");

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

temp[i] = bt[i];

}

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

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 %d\t\t\t %d\n", 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/NOP;

avg\_tat = tat/NOP;

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

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

return 0;

}

**OUTPUT:**

Total number of process: 3

Enter the Arrival and Burst time of the Process[1]

Arrival time is: 0

Burst time is: 24

Enter the Arrival and Burst time of the Process[2]

Arrival time is: 0

Burst time is: 9

Enter the Arrival and Burst time of the Process[3]

Arrival time is: 0

Burst time is: 6

Enter the Time Quantum for the process: 3

Process No Burst Time TAT Waiting Time

Process No[3] 6 18 12

Process No[2] 9 24 15

Process No[1] 24 39 15

Average Turn Around Time: 14.000000

Average Waiting Time: 27.000000