

Ex. No.: 7d

Date 23.4.24

ROUND ROBIN SCHEDULING

Aim:

To implement the Round Robin (RR) scheduling technique

Algorithm:

1. Declare the structure and its elements.
2. Get number of processes and Time quantum as input from the user.
3. Read the process name, arrival time and burst time
4. Create an array `rem_bt[]` to keep track of remaining burst time of processes which is initially copy of `bt[]` (burst times array)
5. Create another array `wt[]` to store waiting times of processes. Initialize this array as 0.
6. Initialize time : $t = 0$
7. Keep traversing the all processes while all processes are not done. Do following for i^{th} process if it is not done yet.
 - a- If $\text{rem_bt}[i] > \text{quantum}$
 - (i) $t = t + \text{quantum}$
 - (ii) $\text{bt_rem}[i] = \text{quantum};$
 - b- Else // Last cycle for this process
 - (i) $t = t + \text{bt_rem}[i];$
 - (ii) $\text{wt}[i] = t - \text{bt}[i]$
 - (iii) $\text{bt_rem}[i] = 0;$ // This process is over
8. Calculate the waiting time and turnaround time for each process.
9. Calculate the average waiting time and average turnaround time.
10. Display the results.

Program Code:

```
#include <stdio.h>
int main()
{
    int, limit, total = 0, X, countes = 0,
    time - quantum;
    int wait-time = 0, turnaround-time = 0,
    arrival-time [10], burst-time [10], temp [10],
```

```

float average_wait_time, average_turnaround_time;
printf ("Enter no. of processes");
scanf ("%d", &limit);
x = limit;
for (i=0; i<limit; i++)
{
    printf ("Enter details of process [%d]", i+1);
    printf ("Arrival Time : t");
    scanf ("%d", &arrival_time[i]);
    printf ("Burst Time : t");
    scanf ("%d", &burst_time[i]);
    temp[i] = burst_time[i];
}

printf ("Enter Time Quantum");
scanf ("%d", &time_quantum);
printf ("process ID TT Burst Time Turnaround Time Waiting Time");
for (total=0, i=0; a!=0;)
{
    if (temp[i]<=time_quantum &&
        temp[i]>0)
    {
        total = total + temp[i];
        temp[i]=0;
        cout<<"50";
    }
}

```

Output: Enter total no. of processes : 4

Enter details of Process [1]

Arrival Time : 0

Burst Time : 4

Enter details of Process [2]

AT : 1

DT : 7

Process [3]

AT : 2

BT : 5

Process [4]

AT : 3

BT : 6

Enter time quantum : 3

Process ID	BT	TAT	WT
1	4	13	9
3	5	16	11
4	6	18	12
2	7	21	14

$$\text{Avg WT} = 11.5$$

$$\text{Avg TAT} = 17.0$$

51

Result: The above commands are executed successfully

23/12/2020