Implementation of Round Robin Scheduling

Perumalla Dharan
AP21110010201

Study in detail about round-robin scheduling.
Implement a C++ program to simulate Round Robin scheduling. Read the burst time and time quantum of N processes and simulate the Round Robin scheduling.

Expected output:

- **The Sequence of process IDs based on the execution order (Same as in the Gantt chart)**
- **A table that demonstrates the response time, completion time, wait time, and turnaround time.**
- **Find the average turnaround time and wait time.**

Code

```
#include <iostream>
using namespace std;

int time_slice;

struct process
{
    int id, burst_time, comp_time, wait_time, tat_time,
resp_time;
};

void round_robin(process pro[], int n)
{
    int time_taken = 0;
    int total = 0;
    int remaining_burst_time[n];
    for (int i = 0; i < n; i++)
    {
```

```
        remaining_burst_time[i] = pro[i].burst_time;
        total += pro[i].burst_time;
    }

    cout << endl
         << "Gant Chart of the given processes" << endl;
    cout << "Pid\t"
         << "Remaining Time\t"
         << "Start Time\t"
         << "End Time" << endl;
    cout << "-------------------------------------------"
<< endl;

    while (time_taken < total)
    {
        for (int i = 0; i < n; i++)
        {
            if (remaining_burst_time[i] == pro[i].burst_time)
                pro[i].resp_time = time_taken;
            if (remaining_burst_time[i] > 0)
            {
                cout << pro[i].id << "\t   " <<
remaining_burst_time[i] << "\t\t   " << time_taken << "\t\t
";

                if (remaining_burst_time[i] <= time_slice)
                {
                    time_taken += remaining_burst_time[i];
                    pro[i].comp_time = time_taken;
                    remaining_burst_time[i] = 0;
                }
                else
                {
```

```
                    time_taken += time_slice;
                    remaining_burst_time[i] -= time_slice;
                }
                cout << time_taken << endl;
            }
        }
    }

    int avg_tat = 0, avg_wt = 0;
    for (int i = 0; i < n; i++)
    {
        pro[i].tat_time = pro[i].comp_time;
        pro[i].wait_time = pro[i].tat_time -
pro[i].burst_time;
        // pro[i].resp_time = pro[i].wait_time;
        avg_tat += pro[i].tat_time;
        avg_wt += pro[i].wait_time;
    }
    cout << endl
         << endl;
    cout << "Average Turn Around Time = " << avg_tat / n <<
endl;
    cout << "Average Wait Time = " << avg_wt / n << endl;
}

int main()
{
    int n;
    cout << "Enter the number of processes " << endl;
    cin >> n;
    cout << "Enter the time slice " << endl;
    cin >> time_slice;
```

```
    process pro[n];
    for (int i = 0; i < n; i++)
    {
        pro[i].id = i + 1;
        cout << "Enter burst time of process " << i + 1 << ":
" << endl;
        cin >> pro[i].burst_time;
        pro[i].comp_time = 0;
        pro[i].tat_time = 0;
        pro[i].wait_time = 0;
        pro[i].resp_time = 0;
    }

    round_robin(pro, n);
    cout << endl
         << endl;
    cout << "Process\t   Burst Time\t   Completion Time   \t
Turnaround Time   \t   Waiting Time   \t   Response Time" <<
endl;
    for (int i = 0; i < n; i++)
    {
        cout << "    " << pro[i].id << "\t     " <<
pro[i].burst_time << "\t\t\t    " << pro[i].comp_time
             << "\t\t\t    " << pro[i].tat_time << "\t\t\t
" << pro[i].wait_time << "\t\t    " << pro[i].resp_time <<
endl;
    }
    return 0;
}
```

Output

```
Enter the number of processes
3
Enter the time slice
2
Enter burst time of process 1:
10
Enter burst time of process 2:
5
Enter burst time of process 3:
8

Gant Chart of the given processes
Pid     Remaining Time  Start Time      End Time
-----------------------------------------------
1         10              0               2
2         5               2               4
3         8               4               6
1         8               6               8
2         3               8               10
3         6               10              12
1         6               12              14
2         1               14              15
3         4               15              17
1         4               17              19
3         2               19              21
1         2               21              23


Average Turn Around Time = 19
Average Wait Time = 12


Process    Burst Time      Completion Time         Turnaround Time          Waiting Time            Response Time
   1         10                      23                      23                      13              0
   2         5                       15                      15                      10              2
   3         8                       21                      21                      13              4
PS E:\SRM\OS\OS LAB>
```

```
Enter the number of processes
4
Enter the time slice
5
Enter burst time of process 1:
21
Enter burst time of process 2:
3
Enter burst time of process 3:
6
Enter burst time of process 4:
2

Gant Chart of the given processes
Pid     Remaining Time  Start Time      End Time
-----------------------------------------------
1          21              0               5
2          3               5               8
3          6               8               13
4          2               13              15
1          16              15              20
3          1               20              21
1          11              21              26
1          6               26              31
1          1               31              32


Average Turn Around Time = 19
Average Wait Time = 11


Process    Burst Time      Completion Time         Turnaround Time          Waiting Time            Response Time
   1         21                    32                      32                    11                     0
   2         3                     8                       8                     5                      5
   3         6                     21                      21                    15                     8
   4         2                     15                      15                    13                     13
PS E:\SRM\OS\OS LAB>
```