**Working with Process Scheduling algorithms**

1. Write a C program to implement Round Robin CPU scheduling algorithm with quantum time – 2 time units. Processes do not arrive at the same time. Display the following results in a table format.

Arrival time and Burst time of every process.

Waiting time and turnaround time of every process.

Average waiting time and turnaround time.

Total idle time of the processor

**Process Burst Time Arrival Time**

P1 12 2

P2 2 0

P3 3 1

P4 7 3

P5 11 5

1. Given the list of processes, their CPU burst times and display/print the Gantt chart for Priority scheduling. Compute the waiting time and turnaround time for each processor. Print the Gantt chart and table of information which contains process name, execution time, waiting time and turnaround time. Finally print the average waiting time and average turnaround time.

**Process Burst Time Priority**

P1 12 2

P2 25 1

P3 13 3

P4 7 5

P5 11 4

#include <iostream>

using namespace std;

void CalculateTurnaround(int a[], int b[], int tmp[], int n, int quantum)

{

    int total = 0, flag = 0, cnt = n, waittime = 0, turnaround = 0;

    cout << "\n Process# \t Burst Time \t Arrival Time \t Turnaround Time \t Waiting Time ";

    for (int i = 0; cnt != 0;)

    {

        if (tmp[i] <= quantum && tmp[i] > 0)

        {

            total += +tmp[i];

            tmp[i] = 0;

            flag = 1;

        }

        else if (tmp[i] > 0)

        {

            tmp[i] -= quantum;

            total += quantum;

        }

        if (tmp[i] == 0 && flag == 1)

        {

            cnt--;

            cout << "\n    " << i + 1 << "\t\t    " << b[i] << "\t\t\t" << a[i];

            cout << "\t\t " << total - a[i] << "\t\t " << total - a[i] - b[i];

            waittime += total - a[i] - b[i];

            turnaround += total - a[i];

            flag = 0;

        }

        if (i == n - 1)

        {

            i = 0;

        }

        else if (a[i + 1] <= total)

        {

            i++;

        }

        else

        {

            i = 0;

        }

    }

    cout << "\n Average Turn Around Time: " << waittime \* 1.0 / n;

    cout << "\n Average Waiting Time    : " << turnaround \* 1.0 / n;

    cout << "\n Total Idle Time         : " << total;

}

int main()

{

    int n, quantum;

    cout << "Enter number of process: ";

    cin >> n;

    cout << "Enter the Time Quantum for the process: ";

    cin >> quantum;

    int a[n], b[n], tmp[n];

    float avg\_wt, avg\_tat;

    cout << "Enter the Burst and Arrival time of the Process:\n";

    for (int i = 0; i < n; i++)

    {

        cin >> b[i] >> a[i];

        tmp[i] = b[i];

    }

    CalculateTurnaround(a, b, tmp, n, quantum);

    return 0;

}