

FCFS.java

```
import java.util.Scanner;
class FCFS {
    // Function to find the waiting time for all
    // processes
    static void findWaitingTime(int processes[], int n,
                                int bt[], int wt[]) {
        // waiting time for first process is 0
        wt[0] = 0;
        // calculating waiting time
        for (int i = 1; i < n; i++) {
            wt[i] = bt[i - 1] + wt[i - 1];
        }
    }
    // Function to calculate turn around time
    static void findTurnAroundTime(int processes[], int n,
                                    int bt[], int wt[], int tat[]) {
        // calculating turn around time by adding
        // bt[i] + wt[i]
        for (int i = 0; i < n; i++) {
            tat[i] = bt[i] + wt[i];
        }
    }
    //Function to calculate average time
    void findavgTime(int processes[], int n, int bt[]) {
        int wt[] = new int[n], tat[] = new int[n];
        int total_wt = 0, total_tat = 0;
        //Function to find waiting time of all processes
        findWaitingTime(processes, n, bt, wt);
        //Function to find turn around time for all processes
        findTurnAroundTime(processes, n, bt, wt, tat);
        //Display processes along with all details
        System.out.printf("Processes Burst time Waiting"
                           +" time Turn around time\n");
        // Calculate total waiting time and total turn
        // around time
        for (int i = 0; i < n; i++) {
            total_wt = total_wt + wt[i];
            total_tat = total_tat + tat[i];
            System.out.printf(" %d ", (i + 1));
            System.out.printf(" %d ", bt[i]);
            System.out.printf(" %d", wt[i]);
            System.out.printf(" %d\n", tat[i]);
        }
        float s = (float)total_wt / (float) n;
        int t = total_tat / n;
        System.out.printf("Average waiting time = %f", s);
        System.out.printf("\n");
        System.out.printf("Average turn around time = %d ", t);
    }
}
//Shortest Remaining Time First(SJF preemptive)
class Process
```

```
{
    int pid; // Process ID
    int bt; // Burst Time
    int art; // Arrival Time

    public Process(int pid, int bt, int art)
    {
        this.pid = pid;
        this.bt = bt;
        this.art = art;
    }
}
```