Assignment 3

```
#include <stdio.h>

int main() {
   int n, i, j, qt, choice, temp, sq = 0, bt[10], tat[10], rem_bt[10], wt[10], at[10]; // Added 'at' array for arrival time
   float awt = 0, atat = 0;

   printf("Enter number of Processes: ");
   scanf("%d", &n);

   printf("Enter the Arrival time and Burst time of processes:\n");
   for (i = 0; i < n; i++) {
      printf("Process %d: ", i + 1);
      scanf("%d %d", &at[i], &bt[i]);
      rem_bt[i] = bt[i];
   }

   printf("Choose Scheduling Algorithm:\n");
   printf("1. Shortest Job First (SJF)\n");
   printf("2. Round Robin (RR)\n");
   printf("Enter your choice: ");
   scanf("%d", &choice);

   switch (choice) {
      case 1: // Shortest Job First
         for (i = 0; i < n; i++) {
            for (j = i + 1; j < n; j++) {
               if (bt[i] > bt[j]) {
                  // Swap burst time
                  temp = bt[i];
```

```
            bt[i] = bt[j];
            bt[j] = temp;

            // Swap arrival time
            temp = at[i];
            at[i] = at[j];
            at[j] = temp;
        }
    }
}

i = 0; // index for processes
j = 0; // index for arrival times
while (i < n) {
    if (rem_bt[i] <= 0) {
        i++;
        continue;
    }

    temp = rem_bt[i];
    sq += temp;
    tat[i] = sq - at[i];
    wt[i] = tat[i] - bt[i];
    rem_bt[i] = 0;
    i++;
}
break;

case 2: // Round Robin
    printf("Enter the Quantum Time: ");
    scanf("%d", &qt);
```

```
        int done = 0;
        i = 0;
        while (!done) {
            done = 1;
            for (i = 0; i < n; i++) {
                if (rem_bt[i] > 0) {
                    done = 0;
                    if (rem_bt[i] > qt) {
                        sq += qt;
                        rem_bt[i] -= qt;
                    } else {
                        sq += rem_bt[i];
                        wt[i] = sq - bt[i];
                        tat[i] = sq - at[i];
                        rem_bt[i] = 0;
                    }
                }
            }
        }
        break;

    default:
        printf("Invalid choice.\n");
        return 1;
}

printf("\nProcess\tArrival Time\tBurst Time\tTurnaround Time\tWaiting Time\n");
for (i = 0; i < n; i++) {
    awt += wt[i];
    atat += tat[i];
    printf("%d\t%d\t\t%d\t\t%d\t\t%d\n", i + 1, at[i], bt[i], tat[i], wt[i]);
}
```

```
    awt /= n;
    atat /= n;
    printf("Average Waiting Time = %f\n", awt);
    printf("Average Turnaround Time = %f\n", atat);

    return 0;
}
```