

```

#include <stdio.h>

int main() {
    int bt[20], p[20], wt[20], tat[20], i, j, n, total = 0, pos, temp;
    float avg_wt, avg_tat;

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

    printf("\nEnter Burst Time for each process:\n");
    for (i = 0; i < n; i++) {
        printf("Process P%d: ", i + 1);
        scanf("%d", &bt[i]);
        p[i] = i + 1; // Process number
    }

    // Sorting burst times in ascending order using selection sort
    for (i = 0; i < n; i++) {
        pos = i;
        for (j = i + 1; j < n; j++) {
            if (bt[j] < bt[pos])
                pos = j;
        }

        // Swap burst time
        temp = bt[i];
        bt[i] = bt[pos];
        bt[pos] = temp;

        // Swap process number to keep track
        temp = p[i];
        p[i] = p[pos];
        p[pos] = temp;
    }

    wt[0] = 0; // First process waiting time is always 0

    // Calculate waiting time
    for (i = 1; i < n; i++) {
        wt[i] = 0;
        for (j = 0; j < i; j++)
            wt[i] += bt[j];

        total += wt[i];
    }

    avg_wt = (float)total / n;
    total = 0;
}

```

```

printf("\n-----");
printf("\n| Process | Burst Time | Waiting Time | Turnaround Time |\n");
printf("-----\n");

// Calculate Turnaround Time and display
for (i = 0; i < n; i++) {
    tat[i] = bt[i] + wt[i];
    total += tat[i];
    printf(" | P%-5d |   %-7d |   %-9d |   %-12d |\n", p[i], bt[i], wt[i], tat[i]);
}

avg_tat = (float)total / n;

printf("-----\n");
printf("\nAverage Waiting Time   = %.2f", avg_wt);
printf("\nAverage Turnaround Time = %.2f\n", avg_tat);

return 0;
}

```

/\* OUTPUT –

Enter the number of processes: 4

Enter Burst Time for each process:

Process P1: 3

Process P2: 4

Process P3: 5

Process P4: 6

-----				
Process	Burst Time	Waiting Time	Turnaround Time	
-----				
P1	3	0	3	
P2	4	3	7	
P3	5	7	12	
P4	6	12	18	
-----				

Average Waiting Time = 5.50

Average Turnaround Time = 10.00

\*/