```
#include <stdio.h>

#define MAX_PROCESSES 10

typedef struct {
    int process_id;
    int arrival_time;
    int burst_time;
    int completion_time;
    int turnaround_time;
    int waiting_time;
} Process;

void fcfs(Process processes[], int n) {
    int total_turnaround_time = 0;
    int total_waiting_time = 0;
    int current_time = 0;

    for (int i = 0; i < n; i++) {
        if (current_time < processes[i].arrival_time)
            current_time = processes[i].arrival_time;

        processes[i].completion_time = current_time + processes[i].burst_time;
        processes[i].turnaround_time = processes[i].completion_time - processes[i].arrival_time;
        processes[i].waiting_time = processes[i].turnaround_time - processes[i].burst_time;

        total_turnaround_time += processes[i].turnaround_time;
        total_waiting_time += processes[i].waiting_time;

        current_time = processes[i].completion_time;
    }


    printf("Process ID | Arrival Time | Burst Time | Completion Time | Turnaround Time | Waiting
Time\n");
    for (int i = 0; i < n; i++) {
        printf("%10d | %12d | %10d | %15d | %15d | %12d\n",
               processes[i].process_id, processes[i].arrival_time, processes[i].burst_time,
               processes[i].completion_time, processes[i].turnaround_time,
processes[i].waiting_time);
    }

      printf("\nAverage Turnaround Time: %.2f\n", (float)total_turnaround_time / n);
    printf("Average Waiting Time: %.2f\n", (float)total_waiting_time / n);
```

```
}

int main() {
    Process system_processes[MAX_PROCESSES], user_processes[MAX_PROCESSES];
    int system_count = 0, user_count = 0;

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

    printf("\nEnter the details for system processes (Process ID, Arrival Time, Burst Time):\n");
    for (int i = 0; i < system_count; i++) {
        system_processes[i].process_id = i + 1;
        printf("Process %d: ", i + 1);
        scanf("%d %d", &system_processes[i].arrival_time, &system_processes[i].burst_time);
    }


    printf("\nEnter the number of user processes: ");
    scanf("%d", &user_count);

    printf("\nEnter the details for user processes (Process ID, Arrival Time, Burst Time):\n");
    for (int i = 0; i < user_count; i++) {
        user_processes[i].process_id = system_count + i + 1;
        printf("Process %d: ", system_count + i + 1);
        scanf("%d %d", &user_processes[i].arrival_time, &user_processes[i].burst_time);
    }

    printf("\n---- System Processes Scheduling ----\n");
    fcfs(system_processes, system_count);

    printf("\n---- User Processes Scheduling ----\n");
    fcfs(user_processes, user_count);

    return 0;
}
```

```
Enter the number of system processes: 3

Enter the details for system processes (Process ID, Arrival Time, Burst Time):
Process 1: 0 10
Process 2: 1 5
Process 3: 2 8

Enter the number of user processes: 2

Enter the details for user processes (Process ID, Arrival Time, Burst Time):
Process 4: 3 6
Process 5: 4 4
```