```c
#include <stdio.h>

void sjf_preemptive(int processes[], int n, int burst_time[])
{
    int remaining_time[n];
    for (int i = 0; i < n; i++)
        remaining_time[i] = burst_time[i];

    int completed = 0, current_time = 0;
    while (completed != n)
    {
        int shortest_job = -1;
        for (int i = 0; i < n; i++)
        {
            if (remaining_time[i] > 0)
            {
                if (shortest_job == -1 || remaining_time[i] < remaining_time[shortest_job])
                    shortest_job = i;
            }
        }

        printf("Executing process %d at time %d\n", processes[shortest_job], current_time);

        remaining_time[shortest_job]--;
        current_time++;

        if (remaining_time[shortest_job] == 0)
        {
            completed++;
            printf("Process %d completed at time %d\n", processes[shortest_job], current_time);
        }
    }
}

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

    int processes[n];
    int burst_time[n];
    printf("Enter the burst time for each process:\n");
    for (int i = 0; i < n; i++)
    {
        printf("Process %d: ", i + 1);
        scanf("%d", &burst_time[i]);
        processes[i] = i + 1;
    }

    sjf_preemptive(processes, n, burst_time);

    return 0;
}
```