Burak Bozdağ 150170110

BLG 335E Analysis of Algorithms

Homework 2 Report

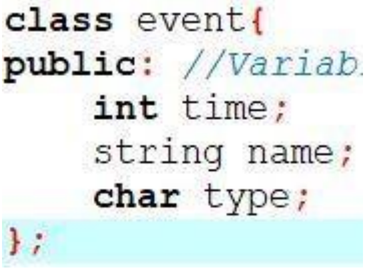
Introduction

In this homework, I implemented an event-scheduler as a MIN-HEAP.

Storing Events

The event scheduler starts by reading the list of events from an event file whose name is supplied as a command line parameter. Event scheduler creates an event MIN-HEAP using the firing times of the events as keys. Each listed event in the events file produces two events: an event at the start time and an event at the end time. Therefore, the heap nodes store event type, either start or end, as well as the event name.

I implemented a class that is used for storing events with their properties such as time, name and type.



```
class event{
public: //Variab
    int time;
    string name;
    char type;
};
```

Figure 1 Class for events

Processing Events

There is a virtual clock in the system which starts from 0 and ticks by 1-time unit. At each clock tick, event scheduler checks the top event in MIN-HEAP to see whether there is a scheduled event at that time or not:

- If there is no scheduled event for that time, the program prints out:

TIME **T**: NO EVENT

where **T** is the current time.

- If there is a START event for the scheduled time, then the program prints out:

TIME **T**: **EVENT-NAME** STARTED

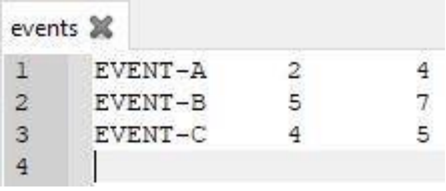
where T is the current time and **EVENT-NAME** is the name of the event. Afterwards, the event is removed from the event MIN-HEAP.

- If there is an END event for the scheduled time, then the program prints out:

TIME T: **EVENT-NAME** ENDED

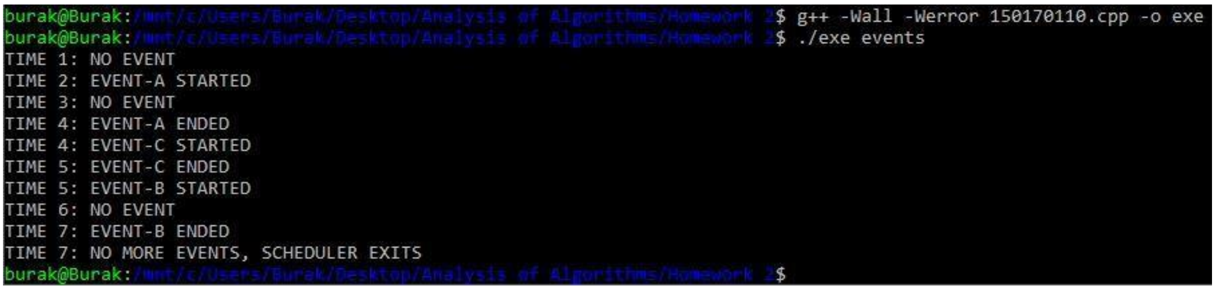
where **T** is the current time and **EVENT-NAME** is the name of the event. Afterwards, the event is removed from the event MIN-HEAP.

An Example Output



```
EVENT-A      2        4
EVENT-B      5        7
EVENT-C      4        5
```

Figure 2 Input file

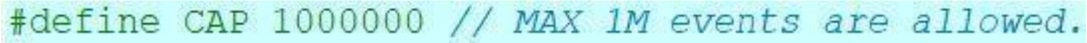


```
burak@Burak:/mnt/c/Users/Burak/Desktop/Analysis of Algorithms/Homework 2$ g++ -Wall -Werror 150170110.cpp -o exe
burak@Burak:/mnt/c/Users/Burak/Desktop/Analysis of Algorithms/Homework 2$ ./exe events
TIME 1: NO EVENT
TIME 2: EVENT-A STARTED
TIME 3: NO EVENT
TIME 4: EVENT-A ENDED
TIME 4: EVENT-C STARTED
TIME 5: EVENT-C ENDED
TIME 5: EVENT-B STARTED
TIME 6: NO EVENT
TIME 7: EVENT-B ENDED
TIME 7: NO MORE EVENTS, SCHEDULER EXITS
burak@Burak:/mnt/c/Users/Burak/Desktop/Analysis of Algorithms/Homework 2$
```

Figure 3 Output from the terminal

Important Notes

- My code allows max. 1M events as default. To change it, you must change the source code before compiling.



```
#define CAP 1000000 // MAX 1M events are allowed.
```

Figure 4 Change CAP according to your needs.

- After defining capacity, you can compile the code and run the program as written in readMe.txt file.

EOF