

DCP1203–HW11

LWC Railway company is planning to build a new train station, thus a electronic timetable showing all the trains is needed. You are asked to implement such a timetable. Each line of the timetable contains:

- a) The number of the train (4 digits, each ranging from 0~9, first digit will not be 0)
- b) The destination of the train (20 characters long at most)
- c) The departure time of the train (00:00~23:59)
- d) The arrival time of the train (00:00~23:59)

For example, a train numbered 3051 heading toward Moscow. This train is leaving at 9:30 and will arrive Moscow at 11:43. The timetable will show:

| Train Number | Destination | Departure Time | Arrival Time |
|--------------|-------------|----------------|--------------|
| 3051 | Moscow | 9:30 | 11:43 |

Your program should contain **5 functions** that will be mentioned later. Every time the program starts, the first thing is identify which action does the user want to do. **Number 1~5 refers to the 5 functions**, and **0 will terminate the program**. Before 0 is entered, **DON'T TERMINATE THE PROGRAM**. It has to execute the instruction given, and **return to the state** where the user enters the instruction.

Remember to **print out the whole table after execution** of each instruction, and **line them up** tidily. The table should be **sorted by the departure time** of the trains. If the timetable is empty, then just print out the indexes.

Please implement the program using **“linked list”**. The introduction to linked list can be found in the end of this document. Here are some suggestions:

1. Use structure containing 1 character array, 3 integers(5 would also work), and 2 pointers to this structure.
2. The program may contain the following functions:
 - a. arriving() -Add the train just arrived to the table
 - b. leaving() -Remove the train just left
 - c. search() -Search for certain train by its number
 - d. insert() -Insert a train
 - e. delete() -Delete a train
 - f. printTable() -Print out the whole table

1. The first function is adding the information of the train that just arrived at the station.

A new line will be added to the bottom of the timetable if a train has arrived. For example, this train has just stopped by the platform:

| Train Number | Destination | Departure Time | Arrival Time |
|--------------|-------------|----------------|--------------|
| 6851 | Berlin | 6:31 | 8:19 |

The following picture shows the input and output (assuming the original timetable is empty):

```
Please enter the instruction: 1
---Please enter the arriving train information---
Train number: 6851
Destination: Berlin
Departure time: 6:31
Arrival time: 8:19

Train Number      Destination      Departure Time  Arrival Time
      6851              Berlin              6:31           8:19

Please enter the instruction:
```

2. The second function is **removing the information of the train that just left by its number**. Suppose the train we added in question 1 just left the platform:

| Train Number | Destination | Departure Time | Arrival Time |
|--------------|-------------|----------------|--------------|
| 6851 | Berlin | 6:31 | 8:19 |

The following picture shows the input and output. Beware that **if the timetable is empty, then removing is forbidden**. If remove is executed when timetable is empty, print out an error message.

```
Please enter the instruction: 2

Train Number      Destination      Departure Time  Arrival Time

Please enter the instruction: 2

There is no train available for departure currently

Please enter the instruction:
```

3. The third function is searching for a particular train by its number. Suppose you are searching this train:

| Train Number | Destination | Departure Time | Arrival Time |
|--------------|-------------|----------------|--------------|
| 1394 | Prague | 12:49 | 13:51 |

The following picture shows the input and output. If there's no train using the entered number, **print out an error message**.

```
Please enter the instruction: 3

---Please enter the train number---
1394
Train Number      Destination      Departure Time    Arrival Time
      1394              Prague              12:49              13:51

Please enter the instruction: 3

---Please enter the train number---
9999

Train 9999 cannot be found

Please enter the instruction:
```

4. The fourth function is inserting a new train into current timetable. Sometimes the railway company needs to schedule more trains to handle an influx of huge crowd.

Suppose the company decided to add this train to the schedule:

| Train Number | Destination | Departure Time | Arrival Time |
|--------------|-------------|----------------|--------------|
| 4771 | Bordeaux | 15:23 | 18:15 |

Please insert the train **according to its departure time**. The following picture shows the input and output:

```
Please enter the instruction: 4

---Please enter the train information you want to insert---
Train number: 4771
Destination: Bordeaux
Departure time: 15:23
Arrival time: 18:15

Train Number      Destination      Departure Time   Arrival Time
      8051          Leipzig              6:19           10:55
      1394          Prague              12:49          13:51
      4771          Bordeaux              15:23          18:15
      2048           Paris              16:55          22:54

Please enter the instruction:
```

5. The fifth function is removing a train from current timetable. Some mechanical problem might popped up just before the train arrive, so the schedule must be cancelled. Suppose this train is temporarily unavailable:

| Train Number | Destination | Departure Time | Arrival Time |
|--------------|-------------|----------------|--------------|
| 1394 | Prague | 12:49 | 13:51 |

The following picture shows the input and output. If there's no train using the entered number, **print out an error message**.

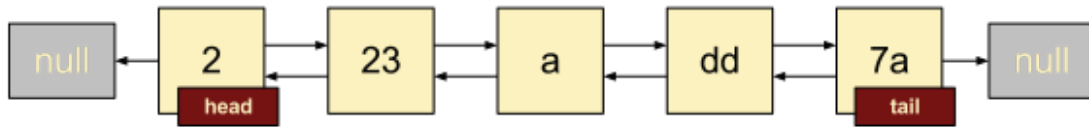
```
Please enter the instruction: 5

---Please enter the train number you want to delete---
1394
Train Number      Destination      Departure Time  Arrival Time
      8051          Leipzig          6:19          10:55
      4771          Bordeaux         15:23          18:15
      2048           Paris          16:55          22:54

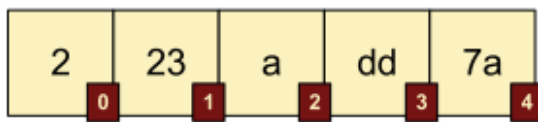
Please enter the instruction: 5
---Please enter the train number you want to delete---
9999
Train 9999 is not in the schedule
Please enter the instruction:
```

Array vs. Linked List

Linked List



Array



Key to linked list:

1. Data is stored as “nodes”
2. Each node is connected to both the node in front and node in back
3. Nodes are connected by pointer to front node and pointer to back
4. Front pointer of head and back pointer of tail point to null

Reference:

1. <https://goo.gl/2z8ZZ3>
2. <https://goo.gl/QUPeCh>