CO222: Programming Methodology

Lab: 10

Deadline: October 11th 2021 @ 11.55PM

In this lab, you need to implement the exact same program as in Lab08. However, the implementation should be done with **linked lists**.

Write a program that can be used to handle a student registration system.

1. The system should keep the following data of each student,
    a. Registration Number
    b. Batch
    c. First Name
    d. Last Name
    e. GPA
2. There should be options to:
    a. Add new students
    b. Delete students
    c. Show the information of a student when his/her registration number is given
    d. Show information about all the students in the system
3. It's fine to make the student registration system volatile. (The data is lost when the program is stopped. No need to write student data to a file or a database.)
4. Internally the program should use a **linked-list** implementation to store student data.
5. The UI should be command-line based. (See the sample UI given.)

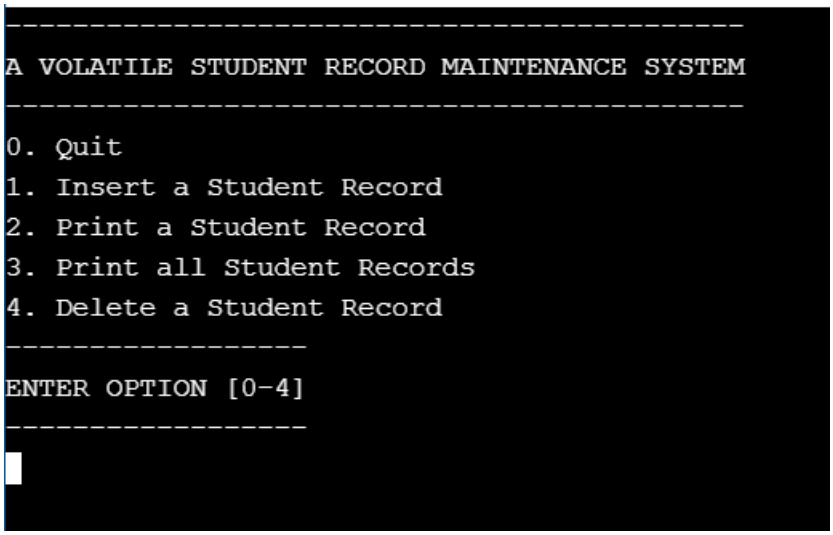

Figure 1: Main UI

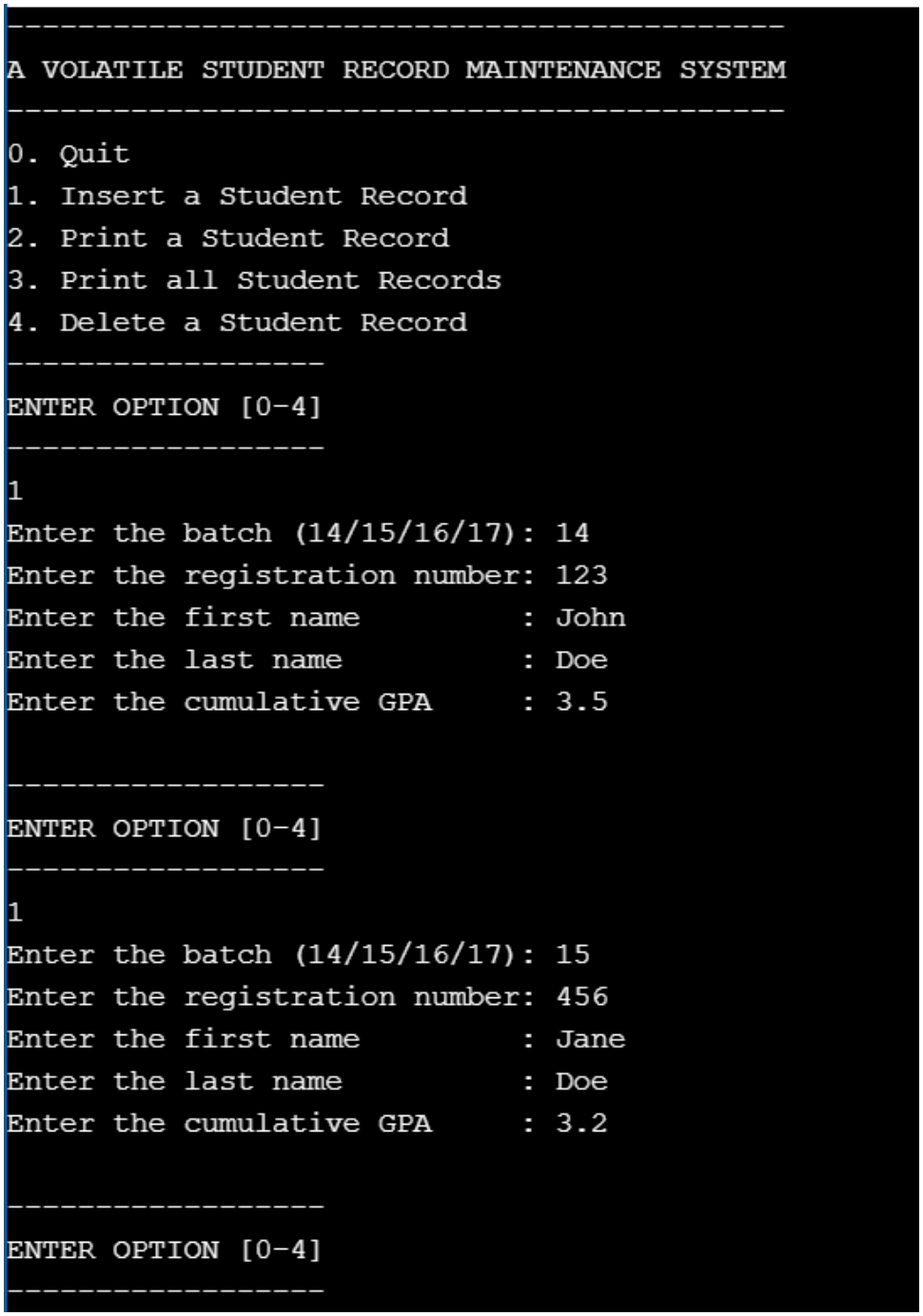

Figure 2: Adding new records

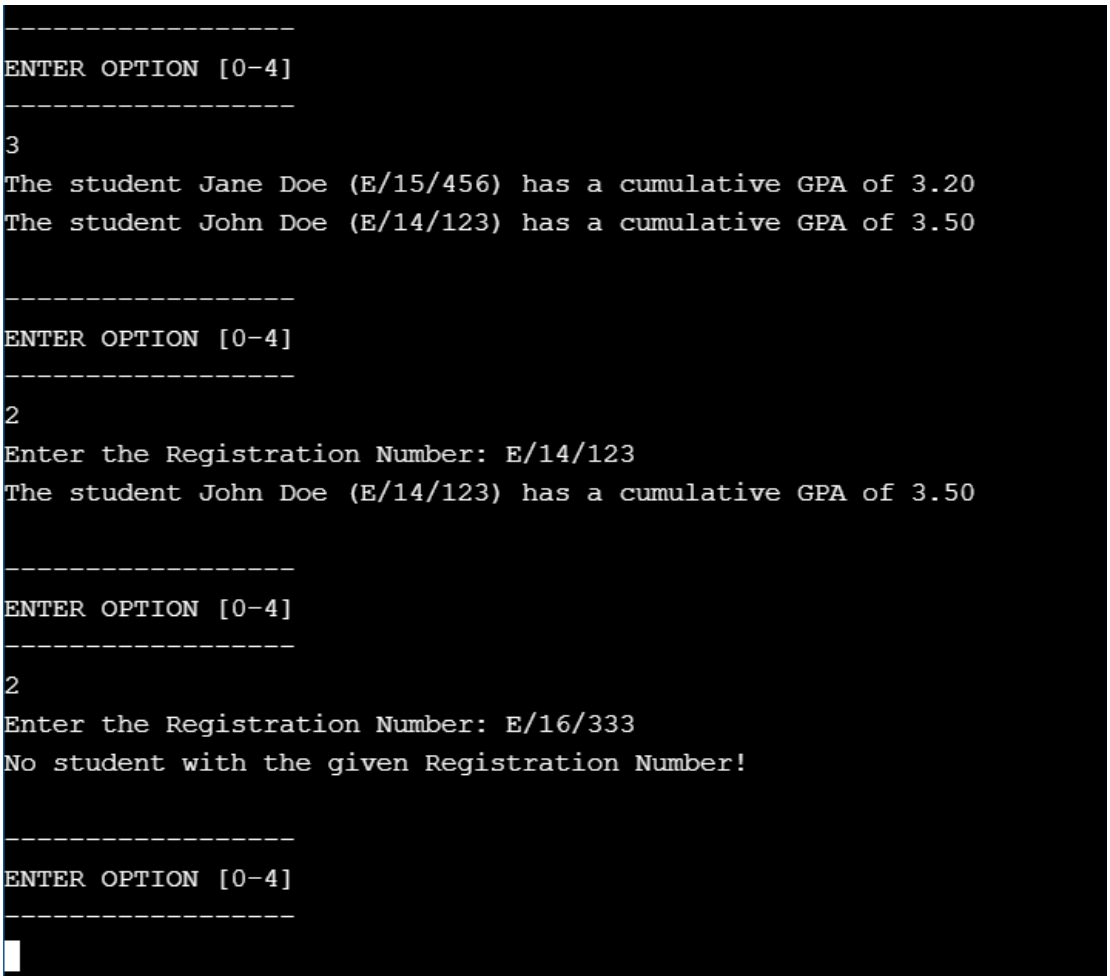

Figure 3: Display Results

```
------------------
ENTER OPTION [0-4]
------------------
1
Enter the batch (14/15/16/17): 14
Enter the registration number: 123
Enter the first name         : John
Enter the last name          : Doe
Enter the cumulative GPA     : 3.5

------------------
ENTER OPTION [0-4]
------------------
1
Enter the batch (14/15/16/17): 15
Enter the registration number: 456
Enter the first name         : Jane
Enter the last name          : Doe
Enter the cumulative GPA     : 3.2

------------------
ENTER OPTION [0-4]
------------------
3
The student Jane Doe (E/15/456) has a cumulative GPA of 3.20
The student John Doe (E/14/123) has a cumulative GPA of 3.50

------------------
ENTER OPTION [0-4]
------------------
```

Figure 3: Order of Print All Records

```
------------------
ENTER OPTION [0-4]
------------------
4
Enter the Registration Number: E/14/123
Delete Successful!

------------------
ENTER OPTION [0-4]
------------------
4
Enter the Registration Number: E/14/123
No student with the given Registration Number!

------------------
ENTER OPTION [0-4]
------------------
```

Figure 4: Deleting a Record

```
-----------------------------------------
A VOLATILE STUDENT RECORD MAINTENANCE SYSTEM
-----------------------------------------
0. Quit
1. Insert a Student Record
2. Print a Student Record
3. Print all Student Records
4. Delete a Student Record
------------------
ENTER OPTION [0-4]
------------------
5
------------------
ENTER OPTION [0-4]
------------------
100
------------------
ENTER OPTION [0-4]
------------------
-5
------------------
ENTER OPTION [0-4]
------------------
0
```

Figure 5: Invalid Option

Instructions

- Start by creating the UI.
- Next, create the structure to store a student record and the linked-list.
    - A structure similar to the following can be used,

```
typedef struct _ {
      int batch;
      int regNo;
      char firstName[20];
      char lastName[20];
      float cGPA;
      struct _* next;

}student_t;
```

- Create separate functions for each operation (Add, Delete, Print) and add them to the UI.
- Write down the answers to the discussion questions below and add them as comments in your code.

- Submit your code to the **Hackerrank Test CO222-2021-Lab 10** **before the deadline** and run all the test cases before submitting.
  Note : The **Hackerrank output will differ from the command line output of GUI** given above. Hence running your code on a command prompt is recommended than using other types of IDEs. If you are using different IDEs, you should be able to handle the output differences on your own.

Discussion

1. How much memory (in bytes) is allocated for your Linked-list with 5 data elements inserted? Show your calculation.
2. At what stage of your program, this memory allocation has happened and when the memory is freed?
3. Explain how deleting values is implemented?
4. Can we add an unlimited amount of student data to this program? If not, what is the limitation?
5. What are the pros and cons of linked lists over arrays?
6. Assume you want a similar system to add exactly 1000 student records at the beginning and after that no additions or deleting. Each record has a unique ID from 0-999. You want to view the student records and modify them. What is the preferred way to implement the system (Array-based or Linked list based)? Explain.