|  |  |
| --- | --- |
| CETN 3010 | PROGRAMMING IN C |

ASSIGNMENT #4 – POINTERS, STRUCTURES and Dynamic Memory Allocation

**Due: Tuesday, Nov 24th, 2015 @ 10:30AM (ie. Before class begins)  
Value: 15% of course mark**

# General Requirements

1. **Define A Structures to hold Student Information**

Your application will need to store information about students. Here is a rough outline of the structure of your data. You will need to create multiple c structures to store the required data:

* + First Name – character array
  + Last Name – character array
  + Date of Birth
    - Day – integer
    - Month – integer (1-12)
    - Year – integer
  + Address
    - Street Address – character array
    - City – character array
  + Final Grade – integer (0-100)

1. **Prompt the end-user to store a number of students at run-time**

When your application begins, the user will be required to tell the application how many students it wishes to enter into the system.

Sample application input:

Welcome to our student information program.

How many students do you wish to enter?: 3

1. **Prompt the end-user to store the data for each student**

The application will then prompt the user to enter in the required information. If will do so for however number of students the user entered at the original prompt.

Sample application input:

Please enter the name for student #1

First Name: John

Last Name: Doe

Please enter the Birth Date for John Doe

Year of birth: 1995

Birth Month (1-12): 4

Birth Day: 25

Please enter the Address for John Doe

Street Address: 123 Main St.

City: Halifax

Please enter the final grade for John Doe

Grade: 95

1. **Provide a list of commands to view entered data**

The application will then provide a list of commands that the program will be able to execute using your entered data. These list of options should be presented to the user repeatedly until the user types ‘q’ or something similar.

Sample application input:

Choose from the following list of options (type q to quit):

1 – Display all students’ names only

2 – Display all students and their dates of birth

3 – Display all students’ data

4 – Display the average grade for all students

5 – Display the student(s) with the highest grade

6 – Add another student

Your choice: 1

John Doe

Jane Smith

Bill Jones

Choose from the following list of options (type q to quit):

1 – Display all students’ names only

2 – Display all students and their dates of birth

3 – Display all students’ data

4 – Display the average grade for all students

5 – Display the student(s) with the highest grade

6 – Add another student

Your choice: 2

John Doe, born April 25, 1995

Jane Smith, born July 15, 1990

Bill Jones, born November 7, 1980

Choose from the following list of options (type q to quit):

1 – Display all students’ names only

2 – Display all students and their dates of birth

3 – Display all students’ data

4 – Display the average grade for all students

5 – Display the student(s) with the highest grade

6 – Add another student

Your choice: 3

John Doe

Birth Date: April 25, 1995

Address: 123 Main St., Halifax

Final Grade: 95

Jane Smith

Birth Date: July 15, 1990

Address: 345 Penny Lane, Dartmouth

Final Grade: 90

Bill Jones

Birth Date: November 7, 1980

Address: 567 Nowhere Blvd, Halifax

Final Grade: 80

Choose from the following list of options (type q to quit):

1 – Display all students’ names only

2 – Display all students and their dates of birth

3 – Display all students’ data

4 – Display the average grade for all students

5 – Display the student(s) with the highest grade

6 – Add another student

Your choice: 4

The average grade for all 3 students is 88.3

Choose from the following list of options (type q to quit):

1 – Display all students’ names only

2 – Display all students and their dates of birth

3 – Display all students’ data

4 – Display the average grade for all students

5 – Display the student(s) with the highest grade

6 – Add another student

Your choice: 5

Student(s) with highest grade

John Doe - 95

1. **Prompt the end-user to add another student to the application**

You’ll need to research and use the **realloc()** function to be able to increase the number of students in your list.

Sample application output:

Choose from the following list of options (type q to quit):

1 – Display all students’ names only

2 – Display all students and their dates of birth

3 – Display all students’ data

4 – Display the average grade for all students

5 – Display the student(s) with the highest grade

6 – Add another student

Your choice: 6

Please enter the name for student #4

First Name: Donny

Last Name: Smith

Please enter the Birth Date for Donny Smith

Year of birth: 199

Birth Month (1-12): 4

Birth Day: 25

Please enter the Address for Donny Smith

Street Address: 123 Main St.

City: Halifax

Please enter the final grade for Donny Smith

Grade: 85

# Submission

You will submit this assignment via time-stamped commits to GitHub.

# Instructions

1. **Don’t forget that a code review is part of this project. You will need to show your code to the instructor in class on the due date while going through an evaluation of the application’s functionality. You will need to explain how the code works and complete the code review part of the rubric.**