# ASSIGNMENT 10

Implement two structures named Student and Course as well as several related operations as explained below:

**Fields of Student:** name, lastName, studentId, mid1Grade, mid2Grade, finalGrade, average

**Fields of Course:** courseName, courseCode, myStudentArray (array of Student structures), currentStudentCount

**Functions:**

void createNewStudent(struct Course *myCourse);

void setGradeOfStudent(struct Course *myCourse);

void findAndDisplayAverage(struct Course *myCourse);

struct Student * findStudentByID(int id, struct Course *myCourse);

void displayAverageOfAllStudents(struct Course *myCourse);

void displayAverageOfStudentsInInterval(struct Course *myCourse);

In main function, you will initialize a single Course structure so that course name and course code become "Fundamentals of Computer Programming" and "CSE114" respectively. In a loop, you will display the following menu:

1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit

For each item in the menu, you will invoke the related method whose prototype is given. Definitions of these functions are explained below:

**createNewStudent:** Prompt the user to enter name, last name and id of the new student. Values entered by the user are assigned to the fields of the student residing in the *myStudentArray* of course variable pointed by *myCourse*. *currentStudentCount* will be updated so that it designates the slot allocated for the student inserted next.

**setGradeOfStudent:** The user will be prompted to enter the id of the student whose grades will be set. ***findStudentByID*** will be invoked to find the student corresponding to the specified id.

**findAndDisplayAverage:** The user will be prompted to enter the id of the student whose average will be computed. ***findStudentByID*** will be invoked to find the student corresponding to the specified id. The average value computed will be assigned to the average field of the student prior to displaying.

**findStudentByID:** Traverses *myStudentArray* in the course variable pointed by *myCourse* to find the student whose id matches the id, which is the parameter of this function. If a matching student exists, ***findStudentByID*** returns its pointer, otherwise it returns NULL.

**displayAverageOfAllStudents:** Traverses *myStudentArray* in the course variable pointed by *myCourse* to display *name*, *last name*, *id* and *average* fields of all students.

**displayAverageOfStudentsInInterval:** The user will be prompted to enter the lowestAvg and the highestAvg. Traverse *myStudentArray* in the course variable pointed by *myCourse* to display *name*, *last name*, *id* and *average* fields of students who have an average in interval [lowestAvg,highestAvg].

**WARNING:**

- Submit only the source file in the format **assignment10_name_surname.c**
- Be sure the extension of your file is **.c**. If you do not know how to check the extension please look at the file (“How to run your code?”) on the COADSYS.
- Do not use any library other than stdio and string (*Hint: strcpy may be useful.*).

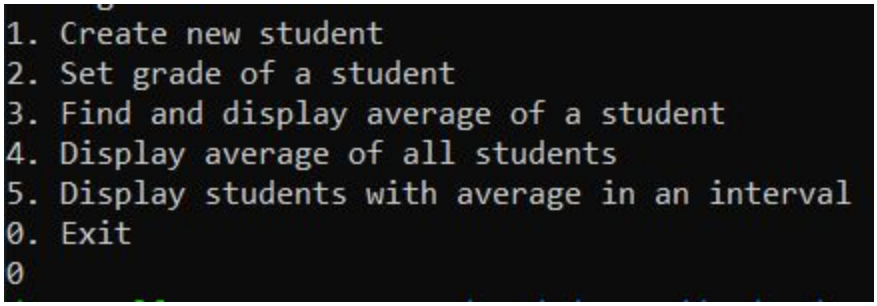
- Give your arrays a fixed size. Assume that no input will be given by the user which exceeds that size. For example, for *myStudentArray[5]*, assume that ***createNewStudent*** will not be called by the user more than 5 times.
- Assume that there will be no invalid tests such as giving a non-existent id to ***setGradeOfStudent***.
- Ask your questions on Coadsys.
- Deniz Tuana Ergonul is responsible for this assignment.

**Examples:**

```
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
1
Name: Marie
Last Name: Curie
Id: 1
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
1
Name: Albert
Last Name: Einstein
Id: 2
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
1
Name: Nikola
Last Name: Tesla
Id: 3
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
2
Id: 1
Midterm 1 Grade: 60
Midterm 2 Grade: 65
Final Grade: 66
```

```
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
2
Id: 2
Midterm 1 Grade: 55
Midterm 2 Grade: 56
Final Grade: 57
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
2
Id: 3
Midterm 1 Grade: 70
Midterm 2 Grade: 60
Final Grade: 40
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
3
Id: 1
Average is: 63.666667
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
3
Id: 2
Average is: 56.000000
```

```
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
3
Id: 3
Average is: 56.666667
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
4
Course Name: Fundamentals of Computer Programming
Course Code: CSE114
Name: Marie
Last Name: Curie
Id: 1
Average: 63.666667
Name: Albert
Last Name: Einstein
Id: 2
Average: 56.000000
Name: Nikola
Last Name: Tesla
Id: 3
Average: 56.666667
1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
5
Lowest Avg: 60
Highest Avg: 100
Course Name: Fundamentals of Computer Programming
Course Code: CSE114
Name: Marie
Last Name: Curie
Id: 1
Average: 63.666667
```


1. Create new student
2. Set grade of a student
3. Find and display average of a student
4. Display average of all students
5. Display students with average in an interval
0. Exit
0