October 27, 2013

Devin Gleason-Lambert

Salem State University

Professor Brikman

CSC325\_01

Assignment 02 \_ Delegates and Lambdas

**Project Specification**

This projects goal is to delve further into the C# language and .NET to learn constructs that we have not seen in other languages before. The use of Delegates and Lambdas lead us away from the easy of earlier assignment where the problems overlapped with concepts we have a firm understanding in. The project takes arrays of different elements (Students and Employees in this example) and has us perform operations on them to determine minimum elements and finding specific elements. The way this is done though is different from ways I have seen in the past, this is because we have the aid of delegates where we are able to specify how we would like to proceed outside of the method doing the operation. Simply we are able to create methods and pass them in order to do the array manipulation.

**Code/Algorithm Design**

There are four components of this program, two structs, one utility class, and a driver class.

**Program**

Program contains the main method which kicks of the execution of the project. Program also has several helper methods, {*populateEmployee, populateStudent, ouputStudents, ouputEmployees, ouputMinEmployee, outputMinStudent, searchStudents, and searchEmployees*}. The methods are done in pairs based on the type of the element, so I will describe what each pair does since the underlying logic is the same.

*Populate* takes an array reference and an int for size, and returns an array of the given size populated with random elements.

*Output* takes an array reference and returns void, its job is to display all the elements of the array it is handed to the Console.

*OutputMin* takes an array reference and returns void, its job is to display the minimum element of the array based on logic from the element and ArrayUtil.

*Search* takes an array reference and returns void, its job is to gather the users input and then search the array for the element based on the users input.

Program also contains further helper methods that you don’t see from the main method that are only called from the initial helper methods {*inputStudent, inputEmployee, NameGenerator, SIDGenerator, and SalaryGenerator}.* These methods fulfill basic functionalities that can be understood simply by their name, therefore I will not talk about them in any more detail.

**ArrayUtil**

*ArrayUtil* contains two static functions, *find* and *findMin*.

*Find* takes parameters, generic array reference, generic target, and delegate Func equals and returns an int. The method goes through the array searching for the target and using logic from Func to determine if elements are equal, if an element is found it returns its index, if not returns -1.

*findMin* takes parameters, generic array, and Func and returns a generic type. findMin searches the array for the minimum element decided by Func, when the search is complete, it returns the generic minimum element.

**Student**

Student contains two attributes, name and sid, a two parameter constructor, and it also contains Func delegates BySID and Equal to provide logic for searching. The ToString and Equals method have also been overridden.

**Employee**

Employee contains two attributes, name and salary, a two parameter constructor, and it also contains Func delegates BySalary, ByName and Equal to provide logic for searching. The ToString and Equals method have also been overridden.

**Test/Input Data Set(s)**

Input for Search Student:

Enter Name:

**Input Results**

abc [valid input]

“ “ [valid input]

“” [valid input]

Enter SID:

**Input Results**

abc [valid input]

“ “ [valid input]

“” [valid input]

Input for Search Employee:

Enter Name:

**Input Results**

abc [valid input]

“ “ [valid input]

“” [valid input]

Enter Salary:

**Input Results**

123 [valid input]

“ “ [Asks for input again]

“” [Asks for input again]

Z [Asks for input again]

**User Manual**

1. After downloading the sources with the project extract the project into a folder.
2. After open a command prompt window and navigate (CD command) to the directory …\CSC325\_Assignment02\CSC325\_Assignment02\bin\Release
3. Type CSC325\_Assignment02.exe and hit enter
4. At this point the program should begin execution.
5. The program will display the Student array and the Employee Array.
6. You may now enter your search term for Student Name.
7. You may now enter your search term for Student SID.
8. The search results will come back.
9. You may now enter your search term for Employee Name.
10. You may now enter your search term for Employee Salary.
11. The search results will come back.
12. You have now come to completion of the program included in this project.