Goal to home project:

1. To have a hand-on experience of programming including coding, compiling, building, running, debugging and testing;
2. To have a idea about the implementation of binary search algorithm;
3. To have a first experience on java language;

Instruction

1. Check if there exists the environment of Java Virtual Machine(JVM). Method: Click Command Prompt and go to DOS console; Input *java* and observe the result from the command. If it reports the help manual, it means JVM has been launched in your local machine.
2. If there is no JVM, download JDK 6 package from http://java.sun.com/javase/downloads/index.jsp and install it in your machine;
3. Download Eclipse software development platform from http://www.eclipse.org/downloads/download.php?file=/eclipse/downloads/drops/R-3.2.2-200702121330/eclipse-SDK-3.2.2-win32.zip
   And install it.
   Articles about eclipse is in website http://www.eclipse.org/articles/

4. Open Eclipse and observe the components in it;

When you want to create a new project in Eclipse, please follow the steps:

- Click "File" in the top menu and then click "New" "Project" submenu;
- In the popped "New Project" frame, select "Java Project" as the wizard; then click "Next >"
- In the "New Java Project" frame, input the unique project name; For example cs2010;
- If you designate the directory of source code, choose "create project from existing source" radio button, click "Browse…" and choose the directory name which store your source code from file chooser dialog; then click "Next>";
- In the next dialog, in source box, click "default package" and check if your source code is shown. Click "Finish"
- In Eclipse's main menu, click "Window", "Open perspective" and then "Java". If it shows you "Package Explorer" in the left panel, it means the above steps are correct.
- Double click your project name, double click the java source code and choose the "Run" submenu in the main menu to run the project.
- It will pop out "Run" dialog if you first run the project; Choose and click "Java Application"; Click "Browse…" button to choose the project name; Click "Search…" button and choose a main class name. Click "Run".
- From main menu, choose "Show view" and then "Console"; later, you can see the results in the console box.

5. In Eclipse
   a) Create a new project with a project name and specified project directory
   b) Build the project

c) Run the project
d) Debug the project with break point
e) Read the binary search code
*) What is the implementation of the binary search algorithm
*) How to insert a number *) How to build a sorted list
f) Try:

1) Replace

```
public SearchArray(int max) {
    a = new long[max]; // create array
    nElems = 0;
}
```

With

```
public SearchArray(int max) {
    a = new long[max]; // create array
    nElems = 10;
}
```

Question: What will happen? Why?

2) Replace int maxSize = 100; with int maxSize = 1000;
Add more elements such as

```
arr.insert(132);
arr.insert(120);
```

Open the shaded comment statements;
Check the running time of both searching approaches;
Question: Which one runs faster when the size of the list is bigger?

3) Extra credit question: In the SearchArray.java, array is the data collection structure. Now we want to check the implementation of the same algorithm for different data structures.

Please download ***LinkedList.java,*** put the source code in the same directory as SearchArray.java, refresh your project in Eclipse, and then read the ***LinkedList.java*** source code. Please think how to implement the sequential search algorithm based on the linked list structure. You could cooperate with your partner.

Please write down your understanding about the linked list structure and your idea about how to implement the sequential search algorithms based on linked list data structure. Properties and methods should be considered for the implementation. It is supposed to have almost the same methods as **SearchArray.java**. Please refer to **SearchArray.java**.

LinkedList.java : (http://www.cs.gsu.edu/~cscqxcx/2010/tools/LinkedList.java)