**TESTS**

SearchById Test:

In the file “SearchTest”, you can see several different test that we conducted on our project. The first of which is opening a file. When you type in the file name, the file is successfully opened. Next, we tested our SearchById method. As you can see, each time we type in an existing id number from the database, the information for that record is returned to the user in an easily readable format. When an id number that is not in the database is put in, such as “00000”, then the user gets an error message and is returned to the menu.

Print Report Test

In the file named “PrintReportTest”, you can see a test of our method that prints the top ten records out for the user. As you can see in the document provided, the report is very nicely formatted and easy for the user to read.

Delete Test

In the file named “DeleteTest”, we have our test for our delete function. As you can see, the user inputs an Id from the database, and then is given the information from that record. The user is then asked if they are sure they want to delete that record. In our test, we chose yes. So the record was deleted. If you take a look in the “DeleteTestImage” file, you will see a finished image of the database, with the record that we deleted now being just a -1. We then go on to search for record 00002, which is found because it is in our database. However, when we search for 00001 it is not found, because that is the record that we deleted.

Modify Test

The file named “ModifyTest” contains our test for our method that modifies a certain field of a record that the user chooses. In this test you can see that we modify all of the fields from record 00001 except the ID. The user cannot choose to modify the ID because it is the primary key. In the “ModifyTestImage” you see the updated fields for the record 00001. As you can see from that image, our input for the Industry field of the record was cut short. This is so that the formatting of the file stays the same to keep our binarySearch function working. This is also implemented when modifying each field, not just the Industry field.

NewEntry Test

The file named “NewEntry Test” contains a single test that was conducted for that method. In this single method, we tested wrong inputs (as you can see in the file) to see how our program would react. When adding a new entry the user is unable to enter an id that is already within the file and is asked to try again. In order for the user to be able to search the newly entered entry, he/she must first exit the database (option 6) and then reopen the database (option 1). This is something we tried to avoid but attempting to code it in our program for the file to automatically close and open so that the user wouldn’t have to but it was unsuccessful. We also tried outfile.flush() but that didn’t work either.

NewDatabase Test

The file, “NewDatabase Test”,like all the other files, this one contains the test we ran for creating a new database. Creating the new database we were successful in, but when searching we were having issues. After having emailed the TAs, we understood that in order for binarySearch to work our new database had to be in the same format as input.txt. Regardless of our attempt to make it so, the database still had issues searching the newly created database however one was created.