|  |  |  |  |
| --- | --- | --- | --- |
| **Functionality Tested** | **Inputs** | **Expected Output** | **Actual Output** |
| Find the union of 2 sets where the result has less than 100 elements. | setA, setB | setC | setC with no duplicates |
| Find the union of 2 sets with some duplicates and where the result has more than 100 elements. | setA, setB | setC with duplicates removed, and upto 100 elements stored | setC with no duplicates and first n objects until setC is full, a message is displayed |
| Find the intersection of two sets | setA, setB | setC which contains Complex objects common in both input sets | setC with common elements |
| Input set from file with <=100 elements | ifstream, setA | setA contains an array of Complex numbers without duplicates | setA contains an array of Complex numbers, no duplicates |
| Input set from screen with <=100 elements | cin, setA | setA contains an array of Complex numbers without duplicates | setA contains an array of Complex numbers, no duplicates |
| Input set from file with >100 elements | ifstream, setA | setA contains an array of upto 100 Complex numbers without duplicates | setA contains the first 100 complex numbers without duplicates, a message is displayed |
| Input format from screen is invalid | cin, setA | Error handled gracefully | Message is displayed and operation not performed. |
| Input set from file with bad filename | ifstream, setA | Error handled gracefully | Message is displayed. Program exits. |
| One set is replaced with another | setA, setB | setA obtains the values of setB with the same size | setA obtains the values of setB with the same size |
| Complex number is assigned to set | setA, c1 | setA becomes a set of one elment, c1 | setA becomes a set of one elment, c1 |
| An element is added to a full set | setA, c1 | operation is not performed | Message is displayed and operation is not performed. |
| Reversibility: complex + set and set + complex | setA, c1 | Both do the same function | both do the same function |
| Items can be removed from sets | setA, c1 | If c1 is found in setA, the object is removed | The object was removed. The last object in the set replaces that object in the array. |

**Assignment 4**

**CSC415**

**Will Rhodes**

**Test Case Design**