**Submission of Your Work**

You need to prepare and submit ONE SINGLE MS Word document to Canvas (in your lab section) as LastName\_FirstName\_Labxy.doc. It must contain:

* Your NAME only on page 1
* For ***each*** question:
  + Specify the question number.
  + Source code. Copy/Paste your final source code. You must include standard “comment header” even if code is provided. *Do Not* paste a snippet of your source code, it must be copy/pasted.
  + Initial test plan. After reading the question requirements, but **before** beginning any coding, create the test case table, below, completed through column Expected Output. Include in your report.
  + Final test plan. Write your program then complete the **test table** with actual output results and include in your report *AFTER* your source code.
  + Output results. Paste in a snippet of output showing results for **every listed test case in your final test plan**, labeled with test case #

Test Table:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test # | Valid / Invalid Data | Description of test | Input Value | Expected Output | Actual Output | Test Pass / Fail |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |

* Add / delete rows from Test Table as necessary
* Modify column widths as necessary
* Test both valid and invalid input
* Test for every output expected
* If failure is an expected output and it happens then that test Passes
* Any test that fails means the program must be fixed so that it passes the test
  + Failing tests need a new test row, ie 1a, 1b, etc, showing corrections from original

You need to write a program to keep inventory for a hardware store five ways.

You will write a program to store the inventory in (1) an unsorted linked list, (2) a sorted linked list, (3) an unsorted array, (4) sorted array, and (5) random access file. Make the array is big enough to store 100 records and singly linked lists.

Each record (struct) will contain:

Record # (int), Tool Name (char [20], *NOT* string), Quantity (int), and cost (double)

Your program should initialize the array and file to 100 empty records (0, spaces, 0,0 respectively).

Then create a menu that allows you to Add a record, Update a record, Delete a record, List a record, and list All tools (the ones that are not empty), and Quit when finished.

Start your inventory with the following information:

|  |  |  |  |
| --- | --- | --- | --- |
| **Record#** | **Name** | **Quantity** | **Cost** |
| 68 | Screwdriver | 106 | 6.99 |
| 17 | Hammer | 76 | 11.99 |
| 56 | Power saw | 18 | 99.99 |
| 3 | Electric Sander | 7 | 57 |
| 83 | Wrench | 34 | 7.5 |
| 24 | Jig Saw | 21 | 11 |
| 39 | Lawn mower | 3 | 79.5 |
| 77 | Sledge hammer | 11 | 21.5 |

Note that the Record# field represents the field to sort on in the array and file.

Perform linear searches on unsorted linked list, sorted linked list, and unsorted array to perform the add, update, delete, and list operations.

Perform direct access to sorted array and file to perform the add, update, delete, and list item operations and a linear search to perform the “list all” operation. For each operation write to an output file to document the operations and the number of comparisons necessary to perform each operation.

For sorted array and file, load data into index equal to the (record number – 1).

After running your program for various inputs write a summary conclusion on the efficiency of performing the five operations in the five ways. Be sure to compare all actions (e.g. insert into list) when coming to your conclusion. Would your conclusion change if you used a doubly-linked list? Why or why not?