# CST8130 – Data Structures

# Assignment 1 – Test Plan

# Written By - Ryan Nguyen: 041137485

## Main menu exception handling

| Description of Test | Input Values | Expected Output | Actual Output / Result | Pass/Fail |
| --- | --- | --- | --- | --- |
| * 1. User inputs valid option | 1 | Do you wish to add a fruit(f), vegetable(v), preserve(p), or juice(j)? | Do you wish to add a fruit(f), vegetable(v), preserve(p), or juice(j)? | Pass |
| * 1. User inputs integer that is not a listed option | 6 | Incorrect value entered | Incorrect value entered | Pass |
| * 1. User inputs non-integer value | e | Incorrect value entered | Incorrect value entered | Pass |

## Option 2: Display current inventory

| Description of Test | Input Values | Expected Output | Actual Output / Result | Pass/Fail |
| --- | --- | --- | --- | --- |
| 1. Display an empty inventory | 2 | Inventory: | Inventory: | Pass |
| 1. Display an inventory with items | 2 | Inventory:  Item: 111 Apple 20 price: $1.00 cost: $0.50 orchard supplier: Apple orchard | Inventory:  Item: 111 Apple 20 price: $1.00 cost: $0.50 orchard supplier: Apple orchard | Pass |

## Option 1: add item to inventory

| Description of Test | Input Values | Expected Output | Actual Output / Result | Pass/Fail |
| --- | --- | --- | --- | --- |
| 1. User chooses option 1 | 1 | Do you wish to add a fruit(f), vegetable(v), preserve(p), or juice(j)? | Do you wish to add a fruit(f), vegetable(v), preserve(p), or juice(j)? | Pass |
| 1. User enters invalid item type | e | Invalid entry. | Invalid entry | Pass |
| 1. User enters valid item type | f | Enter the code for the item: | Enter the code for the item: | Pass |
| 1. User enters not-integer item code | apple | Invalid code  Enter the code for the item: | Invalid code  Enter the code for the item: | Pass |
| 1. User enters valid item code | 123 | Enter the name for the item: | Enter the name for the item: | Pass |
| 1. User enters anything for item name | apple | Enter the quantity for the item: | Enter the quantity for the item: | Pass |
| 1. User enters invalid item quantity | 0 | Invalid entry | Invalid entry | Pass |
| 1. User enters valid item quantity | 10 | Enter the cost of the item: | Enter the cost of the item: | Pass |
| 1. User enters invalid item cost (<= 0) | -1 | Invalid entry | Invalid entry | Pass |
| 1. User enters valid item cost | 2 | Enter the sales price of the item: | Enter the sales price of the item: | Pass |
| 1. User enters invalid sales price (<=0) | 0 | Invalid entry | Invalid entry | Pass |
| 1. User enters valid sales price | 3 | Enter the name of the orchard supplier: | Enter the name of the orchard supplier: | Pass |
| 1. User enters item code that already exists | 123 | Item code already exists | Item code already exists | Pass |
| 1. User attemps to add item to inventory when the inventory is full. | 1 | Inventory at max capacity | Inventory at max capacity | Pass |

## Option 3: buy item(s)

| Description of Test | Input Values | Expected Output | Actual Output / Result | Pass/Fail |
| --- | --- | --- | --- | --- |
| 1. User enters a code that does not exist in inventory | 999 | Code not found in inventory... Error...could not buy item | Code not found in inventory...  Error...could not buy item | Pass |
| 1. User enters valid item code | 111 | Enter valid quantity to buy: | Enter valid quantity to buy: | Pass |
| 1. User enters valid item quantity | 10 | Please select one of the following:  1: Add Item to Inventory  2: Display Current Inventory  3: Buy Item(s)  4: Sell Item(s)  5: To Exit  > | Please select one of the following:  1: Add Item to Inventory  2: Display Current Inventory  3: Buy Item(s)  4: Sell Item(s)  5: To Exit  > | Pass |
| 1. User enters invalid quantity (non-positive integer) | -5 | Invalid quantity...  Error...could not buy item | Invalid quantity...  Error...could not buy item | Pass |
| 1. User attempts to buy items without a record in inventory | 3 | Error...could not buy item | Error...could not buy item | Pass |

## Option 4: sell item(s)

| Description of Test | Input Values | Expected Output | Actual Output / Result | Pass/Fail |
| --- | --- | --- | --- | --- |
| 1. User enters valid item code | 111 | Enter valid quantity to sell: | Enter valid quantity to sell: | Pass |
| 1. User enters invalid item code | 999 | Code not found in inventory...  Error...could not sell item | Code not found in inventory...  Error...could not sell item | Pass |
| 1. User enters valid quantity | 10 | Please select one of the following:  1: Add Item to Inventory  2: Display Current Inventory  3: Buy Item(s)  4: Sell Item(s)  5: To Exit  > | Please select one of the following:  1: Add Item to Inventory  2: Display Current Inventory  3: Buy Item(s)  4: Sell Item(s)  5: To Exit  > | Pass |
| 1. User enters invalid quantity (<= 0) | 0 | Invalid quantity...  Error...could not sell item | Invalid quantity...  Error...could not sell item | Pass |
| 1. User enters quantity greater than amount in storage | 100 | Insufficient stock in inventory... | Insufficient stock in inventory... | Pass |
| 1. User attempts to sell items without a record in inventory | 4 | Error...could not sell item | Error...could not sell item | Pass |

## Option 5: exit

| Description of Test | Input Values | Expected Output | Actual Output / Result | Pass/Fail |
| --- | --- | --- | --- | --- |
| 1. User chooses option 5 | 5 | Exiting… | Exiting... | Pass |

Its expected that you:

<add more tables if you have more features to test>