**AC12001**

Name: …Deji Shote…………………………………………………………………………..

Lab Title: …AC12001 assignment 3: Binary Tree………………………………..……….

Test number/date/version: 30/01/18 ……………………………………….……..

Test Notes: …Tests run on submitted assignment ……………………………

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Description** | **Test Data** | **Expected result** | **Worked?** |
| **Menu Test** | | | |
| View an empty list | Press 2 (input to view list) | No matter what list you choose to view if nothing is added then it will say nothing is added and return you to the start of the menu. | Y (Photo evidence) |
| Remove an item that doesn’t exist. | Press 3 (input to remove item. | It will say that you need to add items first before you can remove them. | Y(Photo evidence) |
| Check the total cost of the binary tree | Press 4(input for binary Tree) | It will say that the cost is 0 because there are no items in the binary tree | Y(photo Evidence) |
| Search for an item that has not been added | Press 5 (input for the Search option) | It will ask for the id of the item the user wants to search for then say that the user needs to add items to the store first. | Y (photo evidence) |
| Press 6 to save the users binary tree to a text file, then load the binary tree and use it. | Press 6 (input for the Save/Load option) | It will save a blank file then load it, if it notices that it is empty it will say that something needs to be added first then redirect the user to the main menu | Y(photo evidence) |
| Exit program | Press 0 | Program should exit | Y |
| **Adding numbers to the binary tree** | | | |
| Add only one number to the binary tree | Input the name, id and price of the item | Program should display name price id and amount of the item | Y |
| Add two numbers to the binary tree | Input the name, id and price of the items | Program should display name price id and amount of the items | Y |
| Add three numbers to the binary tree | Input the name, id and price of the items | Program should display name price id and amount of the items | Y |
| Add four numbers to the binary tree | Input the name, id and price of the items | Program should display name price id and amount of the items | Y |
| Add five numbers to the binary tree | Input the name, id and price of the items | Program should display name price id and amount of the items | Y |
| Add one duplicate into the store with the same id | Enter an item with the same id into the tree twice | The program will notice it and then increase the stock of the item | Y |
| Add an item with the same a different name but same id | Enter an item with the same id into the tree twice | If the ID is the same the program will consider this item a duplicate and just increase the amount | Y |
| Adding an item to a binary tree after loading it from a file | Create a binary tree save it to a file load it right after and try to | The program should have no problem adding this new item to the binary tree loaded | Y  Before the new number was added.    After the new number was added |
| Deleting the root of the binary tree | Create a new binary tree then delete the root and print it in the pre order to see if it replaced properly | It should post the list without anything removed the post the list with the specified item removed after | Y |
| Deleting the leaf of the binary tree | Create a new binary tree then delete the leaf and print it in the pre order to see if it replaced properly | It should post the list without anything removed the post the list with the specified item removed after | Y |
| Deleting Delete a node with one child from the tree | Create a new binary tree then delete the node with one child and print it in the pre order to see if it replaced properly | It should post the list without anything removed the post the list with the specified item removed after | Y |
| Deleting Delete a node with two children from the tree | Create a new binary tree then delete the node with two children from the tree and print it in the pre order to see if it replaced properly | It should post the list without anything removed the post the list with the specified item removed after |  |
| Deleting Delete a node with two children from the tree | Create a new binary tree then delete the node with two children from the tree and print it in the pre order to see if it replaced properly | It should post the list without anything removed the post the list with the specified item removed after | Y |
| Deleting Delete an item that doesn’t exist | Create a new binary tree then try to delete a node with an id that is not in the list | The program should say that the value isn’t in the inventory and then take us back to the menu. | Y |
| Preorder print | Post in binary tree in the preorder way of printing it | The program should post the binary tree in preorder using the ID of the item to sort it | Y |
| Postorder print | Post in binary tree in the Postorder way of printing it | The program should post the binary tree in Postorder using the ID of the item to sort it | Y |
| Inorder print | Post in binary tree in the Inorder way of printing it | The program should post the binary tree in Inorder using the ID of the item to sort it |  |