|  |  |  |  |
| --- | --- | --- | --- |
| 1 | To test the add method, various GroceryItems are attempted to be added to the bag with varying prices/taxable fields. The results are displayed using the print method. | (“test\_add0”,0,false)  (“test\_add1”,1,true)  (“test\_add2”,2,false)  (“test\_add3”,3,true)  (“test\_add4”,4,false) | · test\_add0: $0.00 : tax free  · test\_add1: $1.00 : is taxable  · test\_add2: $2.00 : tax free  · test\_add3: $3.00 : is taxable  · test\_add4: $4.00 : tax free |
| 2 | To test grow, grow is called which should increase the capacity by 5 while also copying the previous bag into the larger bag array. The new capacity is printed along with the results of the print method | The bag from test case #1 | new capacity: 10  · test\_add0: $0.00 : tax free  · test\_add1: $1.00 : is taxable  · test\_add2: $2.00 : tax free  · test\_add3: $3.00 : is taxable  · test\_add4: $4.00 : tax free |
| 3 | To test the salesTax method, the return value of the function is printed wen called on a bag with a known tax total (0.265) | The bag from test case #1 | tax sum of current bag: 0.265 |
| 4 | To test the remove method, GroceryItems are attempted to be removed.   * Case 1: Item does not exist, returns false * Case 2: Item exits, removes it and returns true | \*Case 1  #Instance 1  (“blah”,0,false)  #Instance 2  (“test\_add0”,1,true)  \*Case 2  (“test\_add0”,1,true) | \*Case 1  no parameters match: false  only 'name' matches: false  \*Case 2  all parameters match: true  · test\_add4: $4.00 : tax free  · test\_add1: $1.00 : is taxable  · test\_add2: $2.00 : tax free  · test\_add3: $3.00 : is taxable |

Test Case # Purpose Input Expected Output