

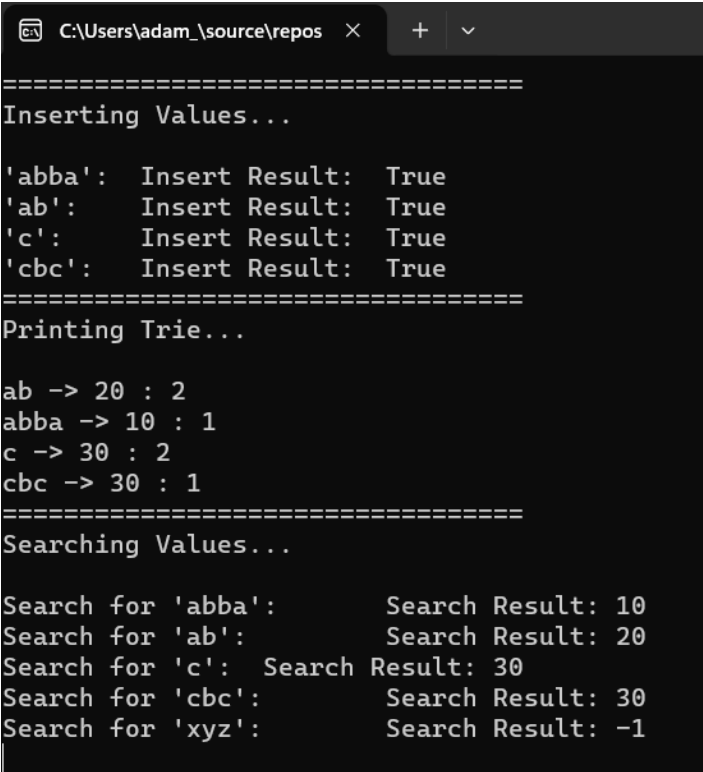
## COIS 3020H: Data Structures and Algorithms II

## Assignment 3 Test Cases

Due Date: December 11<sup>th</sup>, 2023

No.	Description	Input	Test Code	Expected Output	Actual Output
1	Test insertion based on intended use  Also see if print works as intended	("abba", 10); ("ab", 20); ("c", 30);	TestInsert(trie, "abba", 10); TestInsert(trie, "ab", 20); TestInsert(trie, "c", 30); trie.Print(); }  static void TestInsert(RTrie trie, string word, int value) { bool result = trie.Insert(word, value);  Console.WriteLine(\$"{word} ' insert result:\t {result}"); }	'abba' insert result: True 'ab' insert result: True 'c' insert result: True  ab -> 20 : 2 abba -> 10 : 1 c -> 30 : 1	<pre> ===== Inserting Values...  'abba' insert result:      True 'ab' insert result:       True 'c' insert result:        True ab -&gt; 20 : 2 abba -&gt; 10 : 1 c -&gt; 30 : 1   </pre>
2	Test insertion of node with non-unique key and at value is not default	("abba", 10); ("ab", 20); ("c", 30); ("c", 60);	TestInsert(trie, "abba", 10); TestInsert(trie, "ab", 20); TestInsert(trie, "c", 30); TestInsert(trie, "c", 60); .....	'abba' insert result: True 'ab' insert result: True 'c' insert result: True 'c' insert result: False  ab -> 20 : 2 abba -> 10 : 1 c -> 30 : 1	<pre> ===== Inserting Values...  'abba' insert result:      True 'ab' insert result:       True 'c' insert result:        True 'c' insert result:        False ===== Printing Trie...  ab -&gt; 20 : 2 abba -&gt; 10 : 1 c -&gt; 30 : 1   </pre>

3	Test insertion of node with unique key but non-unique value	("abba", 10); ("ab", 20); ("c", 30); ("cbc", 30);	TestInsert(trie, "abba", 10); TestInsert(trie, "ab", 20); TestInsert(trie, "c", 30); TestInsert(trie, "cbc", 30); .....	'abba' insert result: True 'ab' insert result: True 'c' insert result: True 'cbc' insert result: True  ab -> 20 : 2 abba -> 10 : 1 c -> 30 : 2 cbc -> 30 : 1	<pre>===== Inserting Values...  'abba' insert result:      True 'ab' insert result:       True 'c' insert result:        True 'cbc' insert result:      True ===== Printing Trie...  ab -&gt; 20 : 2 abba -&gt; 10 : 1 c -&gt; 30 : 2 cbc -&gt; 30 : 1  </pre>
4	Test search utility for previously inserted pairs	("abba", 10); ("ab", 20); ("c", 30); ("cbc", 30);	TestSearch(trie, "abba"); // Should find the value 10 TestSearch(trie, "ab"); // Should find the value 20 TestSearch(trie, "c"); // Should find the value 30 TestSearch(trie, "cbc"); // Should find the value 30 ..... }  static void TestSearch(RTrie trie, string word) { int result = trie.Search(word);  Console.WriteLine(\$"Search for '{word}':\t Search Result: {result}"); } .....	'abba' insert result: True 'ab' insert result: True 'c' insert result: True 'cbc' insert result: True  ab -> 20 : 2 abba -> 10 : 1 c -> 30 : 2 cbc -> 30 : 1  Search for 'abba': Search Result: 10 Search for 'ab': Search Result: 20 Search for 'c': Search Result: 30 Search for 'cbc': Search Result: 30	<pre>===== Inserting Values...  'abba': Insert Result: True 'ab':   Insert Result: True 'c':    Insert Result: True 'cbc':  Insert Result: True ===== Printing Trie...  ab -&gt; 20 : 2 abba -&gt; 10 : 1 c -&gt; 30 : 2 cbc -&gt; 30 : 1 ===== Searching Values...  Search for 'abba':      Search Result: 10 Search for 'ab':       Search Result: 20 Search for 'c':        Search Result: 30 Search for 'cbc':      Search Result: 30  </pre>

5	Test search utility for non-existent key	<div>Inserted pairs ----- ("abba", 10); ("ab", 20); ("c", 30); ("cbc", 30); ----- Not inserted "xyz"</div>	<div>TestSearch(trie, "abba"); // Should find the value 10 TestSearch(trie, "ab"); // Should find the value 20 TestSearch(trie, "c"); // Should find the value 30 TestSearch(trie, "cbc"); // Should find the value 30 TestSearch(trie, "xyz"); // Should not find, expect -1 ..... }  static void TestSearch(RTrie trie, string word) { int result = trie.Search(word);  Console.WriteLine(\$"Search for '{word}':\t Search Result: {result}"); } .....</div>	<div>'abba' insert result: True 'ab' insert result: True 'c' insert result: True 'cbc' insert result: True  ab -&gt; 20 : 2 abba -&gt; 10 : 1 c -&gt; 30 : 2 cbc -&gt; 30 : 1  Search for 'abba': Search Result: 10 Search for 'ab': Search Result: 20 Search for 'c': Search Result: 30 Search for 'cbc': Search Result: 30 Search for 'xyz': Search Result: -1</div>	<div></div>
---	--	--	--	--	---

6	Test removal utility on inserted pair, then using search to further validate success	<div>Inserted pairs</div> <div>-----</div> <div>("abba", 10);</div> <div>("ab", 20);</div> <div>("c", 30);</div> <div>("cbc", 30);</div> <div>-----</div> <div>Remove pair</div> <div>("abba")</div>	<div>TestRemove(trie, "abba");</div> <div>....</div> <div>}</div> <div>static void TestRemove(RTrie</div> <div>trie, string word)</div> <div>{</div> <div>bool result =</div> <div>trie.Remove(word);</div> <div></div> <div>Console.WriteLine(\$"Remov</div> <div>e '{word}':\t Remove Result:</div> <div>{result}");</div> <div>}</div>	<div>'abba' insert result: True</div> <div>'ab' insert result: True</div> <div>'c' insert result: True</div> <div>'cbc' insert result: True</div> <div></div> <div>ab -&gt; 20 : 2</div> <div>abba -&gt; 10 : 1</div> <div>c -&gt; 30 : 2</div> <div>cbc -&gt; 30 : 1</div> <div></div> <div>Remove 'abba': Remove</div> <div>Result: True</div> <div></div> <div>Search for 'abba': Search</div> <div>Result: -1</div> <div>Search for 'ab': Search</div> <div>Result: 20</div> <div>Search for 'c': Search</div> <div>Result: 30</div> <div>Search for 'cbc': Search</div> <div>Result: 30</div>	<div>=====</div> <div>Inserting Values...</div> <div></div> <div>'abba': Insert Result: True</div> <div>'ab': Insert Result: True</div> <div>'c': Insert Result: True</div> <div>'cbc': Insert Result: True</div> <div>=====</div> <div>Printing Trie...</div> <div></div> <div>ab -&gt; 20 : 2</div> <div>abba -&gt; 10 : 1</div> <div>c -&gt; 30 : 2</div> <div>cbc -&gt; 30 : 1</div> <div>=====</div> <div>Removing Values...</div> <div></div> <div>Remove 'abba': Remove Result: True</div> <div>=====</div> <div>Searching Values...</div> <div></div> <div>Search for 'abba': Search Result: -1</div> <div>Search for 'ab': Search Result: 20</div> <div>Search for 'c': Search Result: 30</div> <div>Search for 'cbc': Search Result: 30</div> <div> </div>
---	--	--	---	--	--

7	Test removal function on non-existent key	<div>Inserted pairs</div> <div>-----</div> <div>("abba", 10);</div> <div>("ab", 20);</div> <div>("c", 30);</div> <div>("cbc", 30);</div> <div>-----</div> <div>Remove</div> <div>"xyz"</div>	<div>TestRemove(trie, "xyz");</div> <div>....</div> <div>}</div> <div><div>'abba' insert result: True</div><div>'ab' insert result: True</div><div>'c' insert result: True</div><div>'cbc' insert result: True</div></div> <div><div>ab -&gt; 20 : 2</div><div>abba -&gt; 10 : 1</div><div>c -&gt; 30 : 2</div><div>cbc -&gt; 30 : 1</div></div> <div><div>Remove 'xyz': Remove</div><div>Result: False</div></div> <div><div>Search for 'abba': Search</div><div>Result: 10</div><div>Search for 'ab': Search</div><div>Result: 20</div><div>Search for 'c': Search</div><div>Result: 30</div><div>Search for 'cbc': Search</div><div>Result: 30</div><div>Search for 'xyz': Search</div><div>Result: -1</div></div>	<div>=====</div> <div>Inserting Values...</div> <div><div>'abba': Insert Result: True</div><div>'ab': Insert Result: True</div><div>'c': Insert Result: True</div><div>'cbc': Insert Result: True</div></div> <div>=====</div> <div>Printing Trie...</div> <div><div>ab -&gt; 20 : 2</div><div>abba -&gt; 10 : 1</div><div>c -&gt; 30 : 2</div><div>cbc -&gt; 30 : 1</div></div> <div>=====</div> <div>Removing Values...</div> <div><div>Remove 'xyz': Remove Result: False</div></div> <div>=====</div> <div>Searching Values...</div> <div><div>Search for 'abba': Search Result: 10</div><div>Search for 'ab': Search Result: 20</div><div>Search for 'c': Search Result: 30</div><div>Search for 'cbc': Search Result: 30</div><div>Search for 'xyz': Search Result: -1</div></div> <div> </div>
---	---	--	---	---

8	Test of the user interface.	<p>Successful RTrie implementation (Using a shortened form of words.txt)</p> <p>-----</p> <p>Print Clear() Prefix(1) Insert(12punch, (nou) -&gt; 12) Search(12punch) Search(10th) Delete(10th) Search(10th) Invalid Option Exit</p>	<p>//User Implementation (code is too big to leave here)</p> <p>// User input: 4, 5, 1, 1, 12punch, nou, 12, 3, 12punch, 3, 10th, 2, 10th, 3, 10th, 8, 7</p>	<pre>// User menu // The entire 36 word trie // User menu // Clear // User Menu 1080 -&gt; /*Some val*/ : 1 10-point -&gt; /*Some val*/ : 1 10th -&gt; /*Some val*/ : 1 11-point -&gt; /*Some val*/ : 1 12-point -&gt; /*Some val*/ : 1 16-point -&gt; /*Some val*/ : 1 18-point -&gt; /*Some val*/ : 1 1st -&gt; /*Some val*/ : 1 // User menu Enter the string to add: 12punch Enter the integer value to add: nou That's not an integer. Enter the integer value to add: 12 Successfully inserted the node // User menu Enter the string to search: 12punch Value: 12 // User menu Enter the string to search: 10th Value: /*Some Value*/ // User menu Enter the string to delete: 10th Successfully deleted the node // User menu Enter the string to search: 10th  // User menu Invalid option. Please enter a valid option. // User menu Exiting the RTrie User</pre>	<p>C:\Users\Alexander (Trent)\source\repos\COIS3020Assignment3\bin\Debu</p> <p>Successfully created the RTrie</p> <p>===== RTrie User Interface =====</p> <ol style="list-style-type: none"><li>1. Insert a string</li><li>2. Remove a string</li><li>3. Search for a string</li><li>4. Print the RTrie</li><li>5. Print the RTrie with a prefix</li><li>6. Clear the command screen</li><li>7. Exit (-1)</li></ol> <p>Enter your choice: 4</p> <p>2 -&gt; 650017414 : 6</p> <p>2,4,5-t -&gt; 749540453 : 1</p> <p>2,4-d -&gt; 781448207 : 1</p> <p>20-point -&gt; 120570463 : 1</p> <p>2D -&gt; 120023333 : 1</p> <p>2nd -&gt; 1292681891 : 1</p> <p>1080 -&gt; 1644873182 : 1</p> <p>10-point -&gt; 136275752 : 1</p> <p>10th -&gt; 1630850725 : 1</p> <p>11-point -&gt; 916679434 : 1</p> <p>12-point -&gt; 1146569726 : 1</p> <p>16-point -&gt; 1765716758 : 1</p> <p>18-point -&gt; 16669994 : 1</p> <p>1st -&gt; 1666468701 : 1</p> <p>&amp;c -&gt; 1643369648 : 1</p> <p>30-30 -&gt; 1473338495 : 1</p> <p>3D -&gt; 1984086594 : 1</p> <p>3-D -&gt; 1041875682 : 1</p> <p>3M -&gt; 334245787 : 1</p> <p>3rd -&gt; 932071594 : 1</p> <p>48-point -&gt; 1353568569 : 1</p> <p>4-D -&gt; 1187663890 : 1</p> <p>4GL -&gt; 1509133388 : 1</p> <p>4H -&gt; 78815602 : 1</p> <p>4th -&gt; 620662413 : 1</p> <p>5-point -&gt; 2009407574 : 1</p> <p>5-T -&gt; 1742128713 : 1</p> <p>5th -&gt; 1501278182 : 1</p> <p>6-point -&gt; 1845256077 : 1</p> <p>6th -&gt; 1416372220 : 1</p> <p>7-point -&gt; 641634925 : 1</p> <p>7th -&gt; 1215539283 : 1</p> <p>8-point -&gt; 1583656815 : 1</p> <p>8th -&gt; 2096618484 : 1</p> <p>9-point -&gt; 593649067 : 1</p> <p>9th -&gt; 1864390371 : 1</p> <p>===== RTrie User Interface =====</p>
---	-----------------------------	---	--	---	---

Interface...  
Exited the User Interface,  
have a good day!

```
C:\Users\Alexander (Trent)\source\repos\COIS3020Assignment3\bin\Debug\COIS3020Assignment3.exe
Command screen cleared.

===== RTrie User Interface =====
1. Insert a string
2. Remove a string
3. Search for a string
4. Print the RTrie
5. Print the RTrie with a prefix
6. Clear the command screen
7. Exit (-1)

Enter your choice: 5

Enter the string to print from: 1
1000 -> 1644873182 : 1
10-point -> 136275752 : 1
10th -> 1630850725 : 1
11-point -> 916679434 : 1
12-point -> 1146569726 : 1
16-point -> 1765716758 : 1
18-point -> 16669994 : 1
1st -> 1666468701 : 1

===== RTrie User Interface =====
1. Insert a string
2. Remove a string
3. Search for a string
4. Print the RTrie
5. Print the RTrie with a prefix
6. Clear the command screen
7. Exit (-1)

Enter your choice: 1

Enter the string to add: 12punch

Enter the integer value to add: nou
That's not an integer.

Enter the integer value to add: 12
Successfully inserted the node.

===== RTrie User Interface =====
1. Insert a string
2. Remove a string
3. Search for a string
4. Print the RTrie
5. Print the RTrie with a prefix
6. Clear the command screen
7. Exit (-1)

Enter the integer value to add: 12
Successfully inserted the node.

===== RTrie User Interface =====
1. Insert a string
2. Remove a string
3. Search for a string
4. Print the RTrie
5. Print the RTrie with a prefix
6. Clear the command screen
7. Exit (-1)

Enter your choice: 3

Enter the string to search: 12punch
Value: 12

===== RTrie User Interface =====
1. Insert a string
2. Remove a string
3. Search for a string
4. Print the RTrie
5. Print the RTrie with a prefix
6. Clear the command screen
7. Exit (-1)

Enter your choice: 3

Enter the string to search: 10th
Value: 1630850725

===== RTrie User Interface =====
1. Insert a string
2. Remove a string
3. Search for a string
4. Print the RTrie
5. Print the RTrie with a prefix
6. Clear the command screen
7. Exit (-1)

Enter your choice: 2

Enter the string to remove: 10th
Successfully removed the node.

===== RTrie User Interface =====
1. Insert a string
2. Remove a string
3. Search for a string
```

```
C:\Users\Alexander (Trent)\source\repos\COIS3020Assignment3\bin\Debug\COIS3020Assignment3.exe
7. Exit (-1)

Enter your choice: 2

Enter the string to remove: 10th
Successfully removed the node.

===== RTrie User Interface =====
1. Insert a string
2. Remove a string
3. Search for a string
4. Print the RTrie
5. Print the RTrie with a prefix
6. Clear the command screen
7. Exit (-1)

Enter your choice: 3

Enter the string to search: 10th
Sorry, but the node could not be found.

===== RTrie User Interface =====
1. Insert a string
2. Remove a string
3. Search for a string
4. Print the RTrie
5. Print the RTrie with a prefix
6. Clear the command screen
7. Exit (-1)

Enter your choice: 8
Invalid option. Please enter a valid option.

===== RTrie User Interface =====
1. Insert a string
2. Remove a string
3. Search for a string
4. Print the RTrie
5. Print the RTrie with a prefix
6. Clear the command screen
7. Exit (-1)

Enter your choice: 7
Exiting RTrie User Interface...
Exited the User Interface, have a good day!
```