Assessed Exercise week 3

Task A :
The static procedure below, will read a text file and display all the words found. Create a BSTree of strings and fill it with the words from a text file. Your console application should also display the number of words in the tree and the height of the tree. There is a text file on moodle you can use to test this.

```
static void readFile(string fileName)
            {
                const int MAX_FILE_LINES = 50000;
                string[] AllLines = new string[MAX_FILE_LINES];

                //reads from bin/DEBUG subdirectory of project directory
                AllLines = File.ReadAllLines(fileName);


                foreach (string line in AllLines)
                {
                    //split words using space , . ?
                    string[] words = line.Split(' ', ',', '.', '?', ';', ':', '!');
                foreach (string word in words)
                    if (word != "")
                        Console.WriteLine(word.ToLower());
                }
            }
```

Task B : Create a console application which provides tests for the **SubTree** method and **Equals** method described in BinarySearchTree Ex4. The test should first construct and display 2 appropriate **AVL trees** and the result of calling each method with those trees. Your tests should cover a non trivial True and False for both methods. When displaying your AVL Trees you should use InOrder and show the height of each as well.

```
public bool Equals(BSTree<T> tree)
 //returns true if this BSTree object contains all the same data as
 //tree with the same structure and ordering of data.

public bool SubTree(BSTree<T> tree)
 //returns true if this BSTree object contains the subtree tree.
 //A subtree of a tree T is a tree consisting of a node in T and all
 // of its descendants in T
```