CSC 228 Data Structures and Algorithms
Fall 2017
Instructor: Dr. Natarajan Meghanathan

Quiz 7 (Take Home: Due on Nov. 6th at 1 PM; Submit through Canvas)
Max. Points: 50

Consider the problem of finding the closest integer to a "test integer" in an array of N positive integers.

For example, in the array [23, 12, 78, 45, 56, 46, 33, 25], the closest integer to "50" is 46.

Brute Force Approach: A brute force approach to determine the closest integer to a test integer would be to scan through the array and determine the absolute value of the difference between the test integer and each of the integers in the array. We keep track of the minimum absolute difference found so far and the corresponding integer in the array. After scanning through the entire array, we can print the corresponding integer that has the minimum absolute difference with the test integer.

Example for the Brute Force Approach (test integer: "50")

Array	[23,	12,	78,	45,	56,	46,	33,	25]
Abs. Diff with "50"	27	38	28	5	6	4	17	25
Min. Abs Diff with "50"	27	27	27	5	5	4	4	4
Estimate of the Closest Integer	23	23	23	45	45	46	46	46

Question 1: What is the asymptotic time complexity (with respect to the number of comparisons) of the brute force approach described above? Justify your answer.

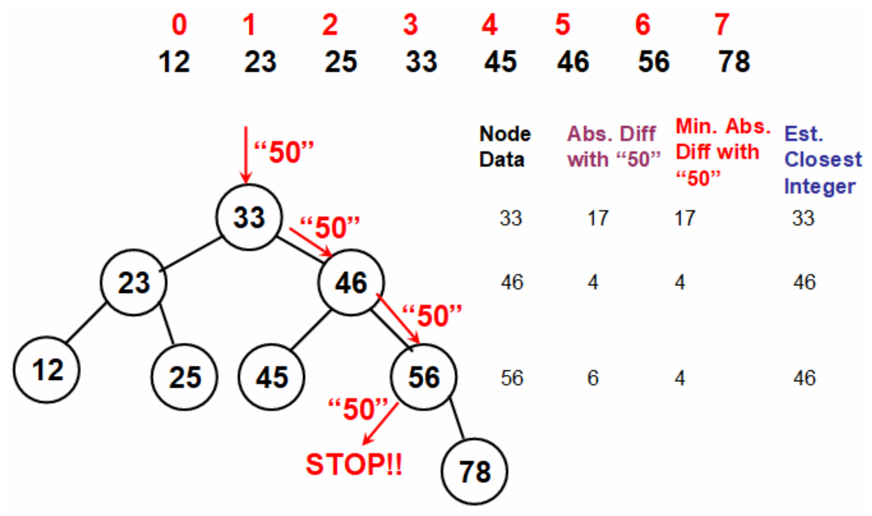
Binary Search Tree-based Approach: A binary search tree (BST)-based approach would be equivalent to searching for the "test integer" in the BST. We start the search process with the root node and proceed to the left sub tree or right sub tree (and recursively thereafter) depending on whether the test integer is less than or greater than the node we are currently at. For every internal node as well as a leaf node, if any, that we come across during the search process, we determine the absolute difference between the test integer and the integer corresponding to the internal node or the leaf node, if any. We keep track of the integer (in the BST) for which the absolute difference with the test integer is the minimum and print the same as the closest integer to the test integer at the end of the search process. If the test integer is indeed the value of an internal node (or a leaf node) during the search process, we simply return the test integer itself as the closest integer and end the search process.

Question 2: What is the asymptotic time complexity (with respect to the number of comparisons) of the BST-based approach? Justify your answer.

Question 3: You are given the code to construct a BST of randomly generated array of integers and search for a "searchKey" using the getKeyIndex( ) function. Your task is to develop the code for the getClosestKey( ) member function (for the Binary Search Tree class) implementing the above algorithm to determine the closest integer in the BST to a "testKey". You could refer to the getKeyIndex( ) function to get an idea of how to search for a particular testKey in the BST. Run the main function as it is, along with the rest of the code (including the getClosestKey( ) function) and capture the screenshot obtained for an array of 12 integers, maximum value for an integer is 99 and the test integer being the last two digits of your J#. The screenshot should include a printout of the array generated and the closest integer to the test integer entered by the user.

Question 4:
Draw a binary search tree for the 12-element array generated to capture your screenshot for Question 3 and illustrate the execution of the algorithm with your testKey as shown in the example below. For each node visited in the BST, you should show the estimated values for the minimum absolute difference between the testKey and the nodes in the BST as well as the estimated values for the closestKey.

Example for the BST-based Approach (test integer: “50”)



0	1	2	3	4	5	6	7
12	23	25	33	45	46	56	78

Node Data	Abs. Diff with “50”	Min. Abs. Diff with “50”	Est. Closest Integer
33	17	17	33
46	4	4	46
56	6	4	46

<u>Submission</u> (through Canvas)
Submit a single word document that contains your answers for questions 1, 2 and 4 as well as the complete code for Question 3 along with a screenshot of the output (as mentioned above).