CS3220  
Assignment 3  
Points: 100

Complete chapter 16, exercise 7 a through d.

* Ex7.
  + Consider an array based implementation of a binary search tree bst. Figure 16-1 presents such a representation for a particular binary search tree
  + A.
    - Depict the array in an array based implementation for the binary search tree in Figure 15-14a of Chapter 15. Assume that tree items are strings

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Jose | Deepak | Mia | Anton | Elisa |  | Qiang |  |  |  |  |  |  |  | Zoe |

* + B.
    - Show the effect of each of the following sequential operations on the array in part *a* of this exercise
      * Bst.add(“Doug”)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Jose | Deepak | Mia | Anton | Doug |  | Qiang |  |  |  | Elisa |  |  |  | Zoe |

* + - * Bst.remove(“Mia”)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Jose | Deepak | Qiang | Anton | Doug |  | Zoe |  |  |  | Elisa |

* + - * Bst.remove(“Deepak”)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Jose | Doug | Qiang | Anton | Elise |  | Zoe |

* + - * Bst.add(“Samara”)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Jose | Doug | Qiang | Anton | Elise |  | Samara |  |  |  |  |  |  |  | Zoe |

* + C.
    - Repeat parts *a* and *b* of this exercise for the tree in Figure 15-14c
      * Array depiction for *c*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Qiang | Jose | Zoe | Deepak | Mia |  |  | Anton | Elisa |

* + - * Add “Doug”

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Qiang | Jose | Zoe | Doug | Mia |  |  | Deepak |  |  |  |  |  |  |  | Anton | Elise |

* + - * Remove “Mia”

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Qiang | Jose | Zoe | Doug |  |  |  | Deepak |  |  |  |  |  |  |  | Anton | Elise |

* + - * Remove “Deepak”

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Qiang | Jose | Zoe | Doug |  |  |  | Anton | Elise |

* + - * Add “Samara”

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Qiang | Jose | Samara | Doug |  |  | Zoe | Anton | Elise |

* + D.
    - Write an in-order traversal algorithm for this array based implementation

Cout>> (“the order of the data in the array for the binary search tree is”);

**For**(**int** I = 0; i<= bst.SIZE; i++)

{

**Int** no = bst[i};

**if**(no != NULL)

{

Printf(no);

}

}