Tennessee TECH

CSC 2110: Program 5

PQSort Using PQAVL

Description: Add a T* remove() method to AVLTree. This method returns and removes either the smallest or the largest item in the AVLTree. To make this determination, you will add a bool min_or_max instance variable to AVLTree, set when the tree is constructed. This variable is set to true to remove and return the smallest item in the AVLTree. Implement this method by calling private methods in AVLTree. Your AVLTree should be balanced after each call to remove.

Your AVLTree will also be able to handle items with duplicate search keys if desired. Add two more bool instance variables, allow_duplicates and duplicates_on_left. The duplicates_on_left bool determines whether to go to the left or the right when a duplicate search key is encountered. Modify insertItem to handle duplicate search keys using these two bool variables.

PQAVL: The PriorityQueue (PQ) ADT is defined as follows:

- PQAVL(bool min_or_max, int (*compare_item) (T* item_1, T* item_2)) //constructor
- ~PQAVL() //destructor
- bool pqIsEmpty()
- void pqInsert(T* item)
- T* pqRemove() //removes and returns the smallest or largest item in the priority queue

Implement the PQ ADT using your AVLTree (PQAVL). PQAVL should be templated. Make sure that you test PQAVL with duplicates.

PQSort: Write a PQSort class with the following methods:

- static T** pqSort(T** items, int num_items, bool min_or_max, int (*compare_item) (T* one, T* two)) //public method to create and return a new, sorted array in case the original unsorted array must be kept, calls the private method below
- static void _pqSort(T** items, int num_items, bool min_or_max, int (*compare_item) (T* one, T* two)) //sorts the array using PQAVL

Implement PQSort using PQAVL. Your sort methods should be templated.

PQSortDriver: Thoroughly test your PQSort implementation. Make sure that you test PQSort with duplicates.

Fully document PQAVL and PQSort:

- Preconditions and Postconditions

- Class level comments

Starting Files: Files I provide as a starting point can be found in Program 5 Files in the Common Files section on Piazza