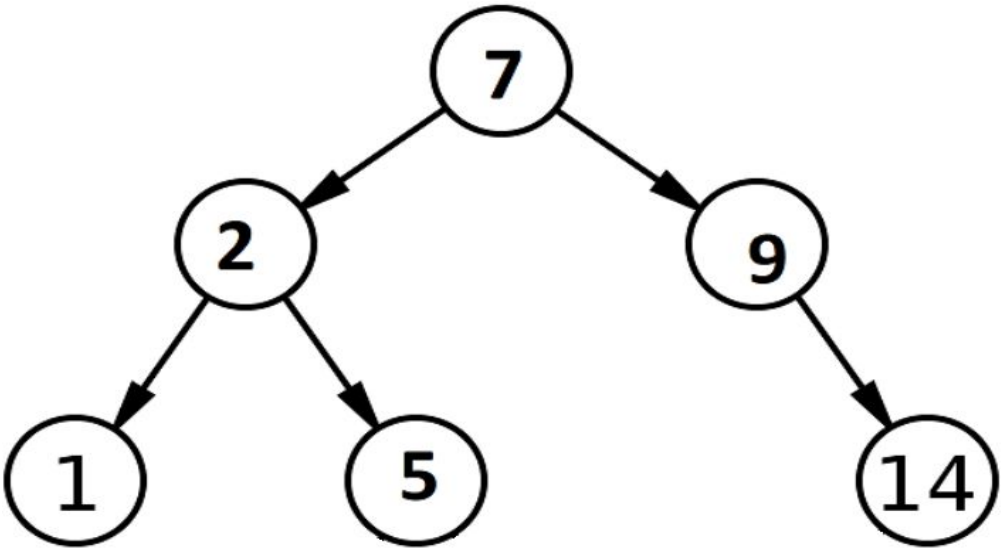
Generic BinarySearchTree

April 27, 2020



Abstract

This Project is about the implementation of Generic BinarySearchTree library built using C++. All necessary Operator overloading is implemented here. Implementation is quite similar to any normal c++ library module. This class also has an Iterator which traverses trees by level order. Each Node is represented by the private class named TreeStruct . This class is a friend of BinaryTree i.e. main class , hence users cannot modify the TreeStruct class.
All the necessary operators are overloaded like equality (==, !=) , assignment (= both copy and move), concatenation(+)

There five different ways user can declare the object

1. *Passing collection of objects, where each objects represents a node in the tree(with or without rank)*

2. *Passing collection of some type and entire collection which act as node(with or without rank)*
3. *Initialising empty tree*

Here rank is provided when object does not provide '<' operator, This kind of objects uses such rank to to create Tree

There is function called BalanceTree() which is used height balance the tree which has time complexity of O(n) and space complexity O(n), n is the no of Node in the tree
Iterator class is provided which traverses by level order.Iterator we implemented here is forward iterator

addNode() and deleteNode() methods are provided which adds and removes the given object from the tree
erase() function used to remove all the available nodes from tree

Members :

1. *Adeesh*
   PES120170959

2. *Manthan BY*
   PES1201701498