|  |  |
| --- | --- |
| **Soal Praktikum**  *Practicum Case* |  |
| T0026  Data Structures |
| **Teknik Informatika**  *Computer Science* | **C1-T0026-RZ01** |
| **Periode Berlaku** Semester Pendek 2016/2017  ***Valid on*** *Compact Semester Year 2016/2017* | **Revisi 00**  *Revision 00* |

## Learning Outcomes

* Demonstrate how to create any learned data structure
* Analyze the usage of data structure in application

## Topic

* Session 06 - Binary Tree

## Sub Topics

* Binary Tree Implementation
* Insert Binary Tree
* Pop 1 node
* Pop All
* Searching in Binary Tree

## Soal

*Case*

Mr. Joe is a manager of a football club named **Blue**. **Blue** is a developing football club, so it has a lot of football player coming in and out. He wants to make the task of managing the player transfer not too complicated. He asks you as a skillful programmer to make a program using the binary tree concept. Here are the descriptions of the program:

* Program consists of 5 menus:

1. View All Player

2. Add Player

3. Remove Player

4. Inorder, Preorder, Postorder

5. Exit and Remove All

* If user chooses **View All Player**, then:

- If there is no data in the tree, show the message **“--- There is No Player in The Tree ---”**

**-** If data is already in the tree, show the player list in this format:

**“Player List:”**

**“- [Player’s Name] ([Player’s Back Number])”**

* If user chooses **Add Player**, then:
  + Ask user to input **player’s name**. Validate that the length of **player’s name** must be **between** **3 and 20 characters**.
  + Ask user to input **player’s back number**. Validate that **the player’s back number** must be **between 1 and 99**.
  + If **the** **player’s back number** already exists, show the message **“ \* Player’s Back Number Cannot be the Same, Please Input Other Number \* ”**

- If tree is still empty, then data will be inserted automatically.

- Otherwise, ask the user to input **the direction** where the data will be placed. Validate that **the** **direction** must be between **“left” and “right”**.

* If the direction chosen is **“left”**, the data will be pushed to the left of current node.
* If the direction chosen is **“right”**, the data will be pushed to the right of current node.

- Maximum tree level is 4. If level is already at maximum, show the message **“--- Maximum Tree Level is 4 ---”**

* + If data has been successfully inputted, show the message **“--- Add Player Success ---”**
* If user chooses **Remove Player**, then:

- If there is no data in the tree, show the message **“--- There is No Player in The Tree ---”**

**-** If data is already in the tree, ask user to input **player’s back number**. Validate that **the player’s back number** must be **between 1 and 99**.

* + If the data can be found, delete the node and its child nodes and show the message **“--- The Player Has Been Removed ---”**
  + If data cannot be found, show the message **“--- The Player Doesn’t Exist ---”**
* If user chooses **Inorder, Preorder, Postorder**, then:

- If there is no data in the tree, show the message **“--- There is No Player in The Tree ---”**

**-** If data is already in the tree, show the **player’s back number** in in-order, pre-order, and post-order.

* If user chooses **Exit and Remove All**, then:

- Delete all data in the linked list.

- Program ends.

**Please run the EXE file to see the sample program.**

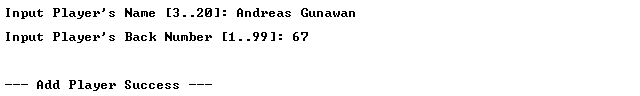
**Print Screen of Main Menu**

****

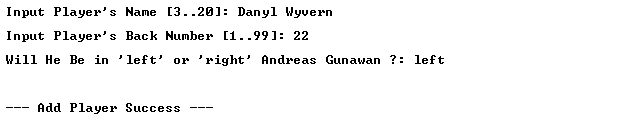
**Print Screen of View All Player Menu (Menu ‘1’)**

****

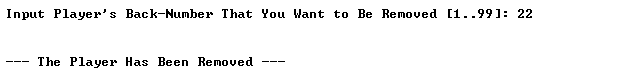
**Print Screen of Add Player Menu (Menu ‘2’) When The Tree Was Still Empty**

****

**Print Screen of Add Player Menu (Menu ‘2’) When The Tree is Not Empty**

****

**Print Screen of Remove Player Menu (Menu ‘3’)**

****

**Print Screen of Inorder, Preorder, Postorder Menu (Menu ‘4’)**

## 