**第46届世界技能大赛移动应用开发项目中国集训队**

**集训日志（选手） 朱姚飞**

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **9-29** | **指导老师** | **刘雄华** |
| **训练任务** | **二叉树遍历** | | | |
| **训练内容：**  二叉树遍历  **训练要求：**  编写 前序遍历中序遍历后续遍历 的 Stack 和 递归版本  **过程记录：**  class Node<T> {    Node? left;    Node? right;    T value;    Node({this.left, this.right, required this.value});  }  class Stack<T> {    List<T> \_list = [];    int push(T e) {      \_list.add(e);      return \_list.length;    }    T pop() {      return \_list.removeLast();    }    int get length => \_list.length;  }  */// 前序遍历 Recursion*  */// Center Left Right*  void CLRRecursion(Node? root) {    if (root != null) {      print(root.value);      CLRRecursion(root.left);      CLRRecursion(root.right);    }  }  */// Stack 版本*  void CLRStack(Node root) {    final stack = Stack<Node>();    stack.push(root);    while (stack.length > 0) {      final next = stack.pop();      print(next.value);      if (next.right != null) {        stack.push(next.right!);      }      if (next.left != null) {        stack.push(next.left!);      }    }  *// print(root.value);*  *// CLRRecursion(root.left);*  *// CLRRecursion(root.right);*  }  */// 中序遍历 Recursion*  */// Left Center Right*  void LCRRecursion(Node? root) {    if (root != null) {      LCRRecursion(root.left);      print(root.value);      LCRRecursion(root.right);    }  }  */// 其他同理 只需改变顺序*  */// 所以省略 Stack 版本*  */// 后序遍历 Recursion*  */// Left Right Center*  void LRCRecursion(Node? root) {    if (root != null) {      LRCRecursion(root.left);      LRCRecursion(root.right);      print(root.value);    }  }  */// 省略 后序遍历 Stack 版本*  */// Deep 获取深度*  int Deep(Node? root, [int deep = 0]) {    if (root != null) {      deep++;      final l = Deep(root.left, deep);      final r = Deep(root.right, deep);      return l > r ? l : r;    }    return deep;  }  */\*\**  *\**  *\* left*  *\*        4*  *\*      /  \*  *\*    2     6     10*  *\*   /          /*  *\*  /         9*  *\* 1        /*  *\*   \    5*  *\*    \ /  \*  *\*     3    8*  *\*     \*  *\*      7*  *\* right*  *\*/*  main(List<String> args) {    final root = Node(      value: 1,      left: Node(        value: 2,        left: Node(          value: 4,          right: Node(            value: 6,          ),        ),      ),      right: Node(        value: 3,        left: Node(          value: 5,          left: Node(            value: 9,            left: Node(value: 10),          ),          right: Node(value: 8),        ),        right: Node(value: 7),      ),    );    print(Deep(root));  }  **分析总结：**  二叉树遍历 | | | | |

**填写人：朱姚飞**