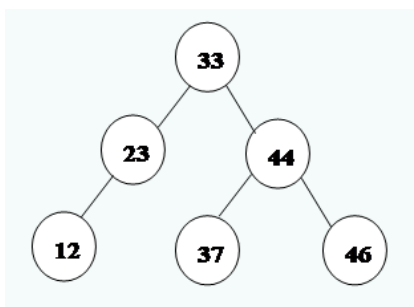

实验四 利用队列实现二叉树的层次遍历（4 课时）

【实验目的】

- 1.理解队列的受限操作（在一端插入，另一端删除）以及先进先出的特性，体会队列的应用场景的特点；
- 2.能利用《程序员实用算法》的程序清单 2-9 和 2-10 中的现有函数来解决实际应用问题。

【实验要求】

应用需求：要求实现：若键盘输入 t，则对当前二叉树进行非递归的层次遍历，并打印出来。（比如，若二叉树为：



则层次遍历后，打印出的结果为“33,23,44,12,37,46”。）

问题描述：利用队列作为数据结构来实现非递归的二叉树层次遍历的函数。

算法思想：

关键：实现非递归的二叉树层次遍历的函数（函数原型为：`int LevelTraBintree(Bintree *t, DoFunc df);`

输入：二叉排序树 t，节点的操作函数 df；

输出：若是空树，则返回 TREE_FAIL；否则，返回 TREE_OK.

算法设计思路：

- 1.分别创建队列和空闲列表；
- 2.为空闲链表分配 QMAX 个结点；
- 3.若二叉树为空，则退出程序；
- 4.将二叉树根节点入队；
- 5.重复执行下列操作，直至队列为空。
 - 1) 将队头节点出队并打印出其值；
 - 2) 若出队节点有左子树，则将该节点的左孩子入队；
 - 3) 若出队节点有右子树，则将该节点的右孩子入队；
- 6.返回成功，程序退出。

【提示】：

- 1.附件中提供了项目文件代码，且已完成相关的数据结构的定义，只需完成函数 LevelTraBintree 的函数体代码即可。可以部分借鉴《程序员实用算法》的程序清单 2-11 中的

`main` 函数代码。

2. 队列和空闲列表中的数据域分别指向 `struct NodeData1` 和 `struct NodeData2` 类型（这两种类型与 `bintree.h` 中定义的 `Mynode` 类型是一致的）。

Note: 请在线提交源码和实验报告，其中实验报告务必按照给定的模板来撰写。