

本节内容

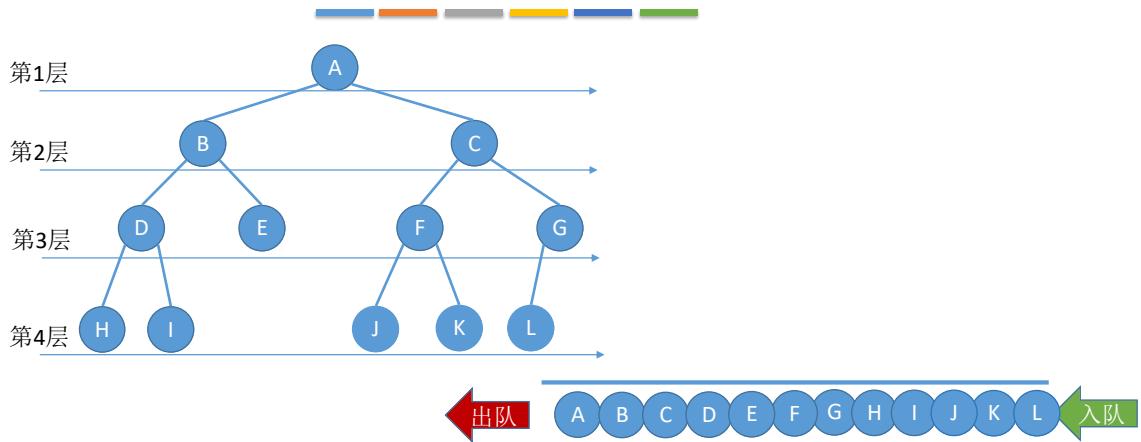
二叉树

层序遍历

王道考研/CSKAOYAN.COM

1

二叉树的层序遍历



算法思想：

- ① 初始化一个辅助队列
- ② 根结点入队
- ③ 若队列非空，则队头结点出队，访问该结点，并将其左、右孩子插入队尾（如果有的话）
- ④ 重复③直至队列为空

王道考研/CSKAOYAN.COM

2

代码实现

算法思想：

- ① 初始化一个辅助队列
- ② 根结点入队
- ③ 若队列非空，则队头结点出队，访问该结点，并将其左、右孩子插入队尾（如果有的话）
- ④ 重复③直至队列为空

```
// 层序遍历
void LevelOrder(BiTTree T){
    LinkQueue Q;
    InitQueue(Q); // 初始化辅助队列
    BiTree p;
    EnQueue(Q, T); // 将根结点入队
    while(!IsEmpty(Q)){ // 队列不空则循环
        DeQueue(Q, p); // 队头结点出队
        visit(p); // 访问出队结点
        if(p->lchild!=NULL)
            EnQueue(Q, p->lchild); // 左孩子入队
        if(p->rchild!=NULL)
            EnQueue(Q, p->rchild); // 右孩子入队
    }
}
```

```
// 二叉树的结点（链式存储）
typedef struct BiTNode{
    char data;
    struct BiTNode *lchild,*rchild;
}BiTNode,*BiTree;

// 链式队列结点
typedef struct LinkNode{
    BiTNode * data; // 存指针而不是结点
    struct LinkNode *next;
}LinkNode;

typedef struct{
    LinkNode *front,*rear; // 队头队尾
}LinkQueue;
```

王道考研/CSKAOYAN.COM

3

知识回顾与重要考点

树的层次遍历算法思想：

- ① 初始化一个辅助队列
- ② 根结点入队
- ③ 若队列非空，则队头结点出队，访问该结点，并将其左、右孩子插入队尾（如果有的话）
- ④ 重复③直至队列为空

王道考研/CSKAOYAN.COM

4