

//5.5.3 节 算法 LevelOrder

// 按层次次序遍历以当前结点为根的树，利用一个辅助队列

```
template<class T>
```

```
void Tree<T> :: LevelOrder ( TreeNode<T> * t )
```

```
{
    AQueue<TreeNode<T>*> q ;           //队列
    if(t!=NULL)                        // 若树不为空树，开始树的层次遍历
    {
        TreeNode<T> *p ;
        q.QInsert ( t );               // 结点 t 入队
        while(! q.IsEmpty ( ))
        {
            q.QDelete ( p );           // 出队一个结点 p
            cout<<p->GetData() <<endl; // 访问结点 p
            p = FirstChild ( p );      // 将结点 p 的所有子结点入队
            while(p!=NULL)
            {
                q.QInsert ( p );
                p = NextBrother ( p );
            }
        }
    }
}
```