

实验九 串里的字符数

目标

理解建立一棵二叉排序树 (BST) 的过程, 进一步熟悉递归算法的设计。

内容

写一个程序, 读入一个字符串, 计算该串中各字符的个数, 并按字符的 ASCII 码升序输出各字符的个数。要求用二叉排序树存放各字符及其个数。树结点结构可包括以下域:

character;

count of the character;

pointer to its left subtree where each node has a lower ASCII code;

pointer to its right subtree where each node has a higher ASCII code;

提示

根据用户输入的字符串, 创建一棵二叉排序树。同时计算 (累加) 串中各字符的出现次数, 将其存在相应的结点中。然后, 以中序遍历的顺序输出树的所有结点。由二叉排序树的性质可知, 这样便可按 ASCII 码升序的顺序输出所有字符。可设计如下函数。

- InsertBST(BSTNode *&bt, char c): Insert a character *c* into a BST *bt*. It will be elegant to design this function using recursion.
- InOrder(BSTNode *bt): Inorder traverse a BST *bt* to print all nodes.
- DestroyBST(BSTNode *bt): Destroy a BST *bt*.

要求

当用户执行此程序, 产生类似下图的显示。

```
Input a string: Hello world!  
Characters and their numbers are:  
    (1)  
    ! (1)  
    H (1)  
    d (1)  
    e (1)  
    l (3)  
    o (2)  
    r (1)  
    w (1)
```