xingwenpeng

 $July\ 29,\ 2014$

${\bf Contents}$

1 数据结构面试题	1
1.1 第一篇	1
1.1.1 把二元查找树转变成排序的双向链表	1
1.1.2 下排每个数都是先前上排那十个数在下排出现的次数 .	3
1 数据结构面试题	
1.1 第一篇	
1.1.1 把二元查找树转变成排序的双向链表	
题目:输入一棵二元查找树,将该二元查找树转换成一个排序的双向较表。要求不能创建任何新的结点,只调整指针的方向	连
#include <stdio.h></stdio.h>	
#include <stdlib.h></stdlib.h>	
struct BSTTreeNode	
{	
int value;	
struct BSTTreeNode *1;	
struct BSTTreeNode *r;	
} ;	
struct BSTTreeNode *head = NULL;	
struct BSTTreeNode*NODE(int item, struct BSTTreeNode *1, struct {	BSTTreeNode *r)
<pre>struct BSTTreeNode *t = malloc(sizeof *t);</pre>	

```
t->value = item;
    t->1 = 1; t->r = r;
    return t;
}
struct BSTTreeNode *insert_node(struct BSTTreeNode *t, int value)
    if (t==NULL) return NODE(value, NULL, NULL);
            (value < t->value)
    t->l = insert_node(t->l, value);
    else if (value > t->value)
    t->r = insert_node(t->r, value);
    return t;
}
void treetodoublelist(struct BSTTreeNode *t)
    static struct BSTTreeNode *pcurrent = NULL;
    t->l = pcurrent;
    if (NULL == pcurrent)
    head = t;
    else
    pcurrent->r = t;
    pcurrent = t;
void in_ord(struct BSTTreeNode *t)
{
    if (t==NULL) return;
    in_ord(t->1);
    treetodoublelist(t);
    in_ord(t->r);
}
void in_ord_show(struct BSTTreeNode *t)
    if (t==NULL) return;
    in_ord_show(t->1);
    printf("%d\n", t->value);
    in_ord_show(t->r);
void show_list(struct BSTTreeNode *h)
{
    struct BSTTreeNode *p;
```

```
if (h == NULL) return;
   for (p = h; p != NULL; p = p->r)
   printf("%d\n", p->value);
}
int main(int argc, char *argv[])
{
   struct BSTTreeNode *root = NULL;
   root = insert_node(root, 10);
   root = insert_node(root, 6);
   root = insert_node(root, 14);
   root = insert_node(root, 4);
   root = insert_node(root, 8);
   root = insert_node(root, 12);
   root = insert_node(root, 16);
   //中序打印排序树
    in_ord_show(root);
   //排序树转换为双向链表
   in ord(root);
   //遍历双向链表
   show list(head);
   return 0;
}
```

1.1.2 下排每个数都是先前上排那十个数在下排出现的次数

给你 10 分钟时间,根据上排给出十个数,在其下排填出对应的十个数要求下排每个数都是先前上排那十个数在下排出现的次数。

```
上排的十个数如下: 【0, 1, 2, 3, 4, 5, 6, 7, 8, 9】
举一个例子,
数值: 0,1,2,3,4,5,6,7,8,9
分配: 6,2,1,0,0,0,1,0,0,0
0 在下排出现了 6 次, 1 在下排出现了 2 次,
2 在下排出现了 1 次, 3 在下排出现了 0 次....
以此类推..
```