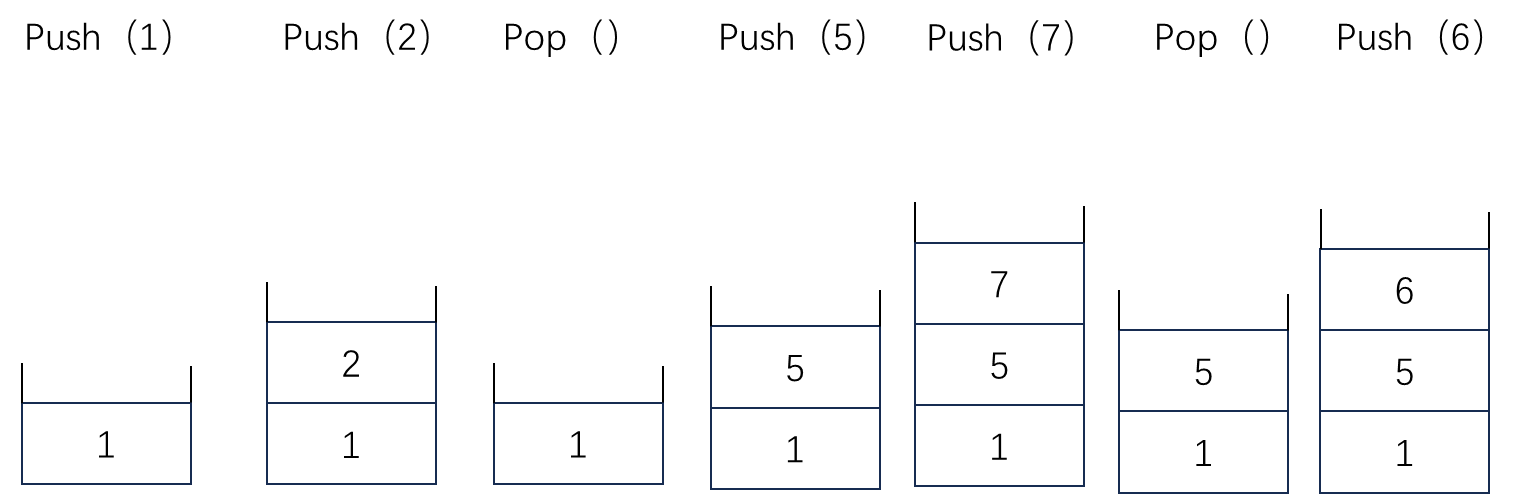
**部分参考答案（部分答案有误，请大家仔细思考）**

1. 应用题
2. 在一个栈中做操作序列Push(1)，Push(2)，Pop，Push(5)，Push(7)，Pop，Push(6)，画出每次操作后，栈的变化情况。

答案：

2、图1是一个带权的无向图，

10

7

9

5

6

7

11

8

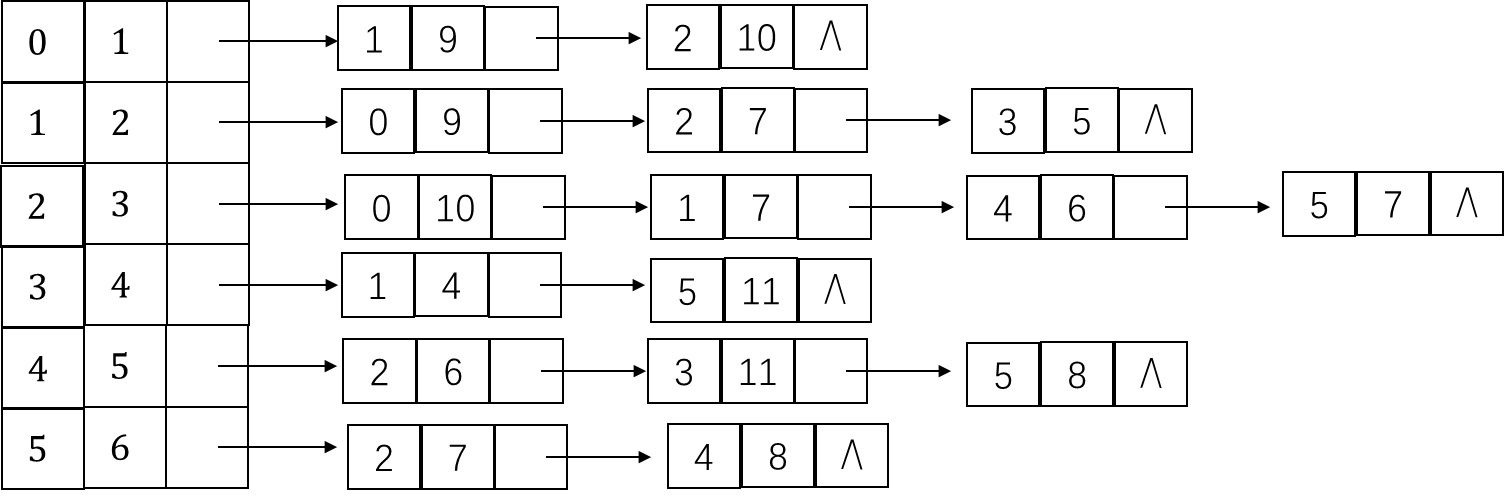
图1

（1）画出该图的邻接矩阵和邻接链表。

（2）采用克鲁斯卡尔算法构造该图的最小生成树，画出最小生成树的过程。

答：（1）邻接矩阵

邻接链表：



（2）

5

5

6

7

5

6

7

5

6

7

7

9

5

6

7

3、已知一组序列为{53，46，32，64，27，48，70，60，75，80，55，63，77，36},

（1）请使用快速排序法对该序列排序，画出每一趟排序结果。

（2）根据（1）的结果，采用折半查找法找到32，画出比较过程。

（3）画出折半查找判定树，并计算查找成功的平均比较次数。

（1）答：下划线的数字表示当前的轴枢

第一趟：36 46 32 48 27 53 70 60 75 80 55 63 77 64

第二趟：27 32 36 48 46 53 70 60 75 80 55 63 77 64

第三趟：27 32 36 48 46 53 70 60 75 80 55 63 77 64

第四趟：27 32 36 48 46 53 70 60 75 80 55 63 77 64

第五趟：27 32 36 46 48 53 70 60 75 80 55 63 77 64

第六趟：27 32 36 46 48 53 70 60 75 80 55 63 77 64

第七趟：27 32 36 46 48 53 64 60 63 55 70 80 77 75

第八趟：27 32 36 46 48 53 55 60 63 64 70 80 77 75

第九趟：27 32 36 46 48 53 55 60 63 64 70 80 77 75

第十趟：27 32 36 46 48 53 55 60 63 64 70 80 77 75

第十一趟：27 32 36 46 48 53 55 60 63 64 70 80 77 75

第十二趟：27 32 36 46 48 53 55 60 63 64 70 75 77 80

第十三趟：27 32 36 46 48 53 55 60 63 64 70 75 77 80

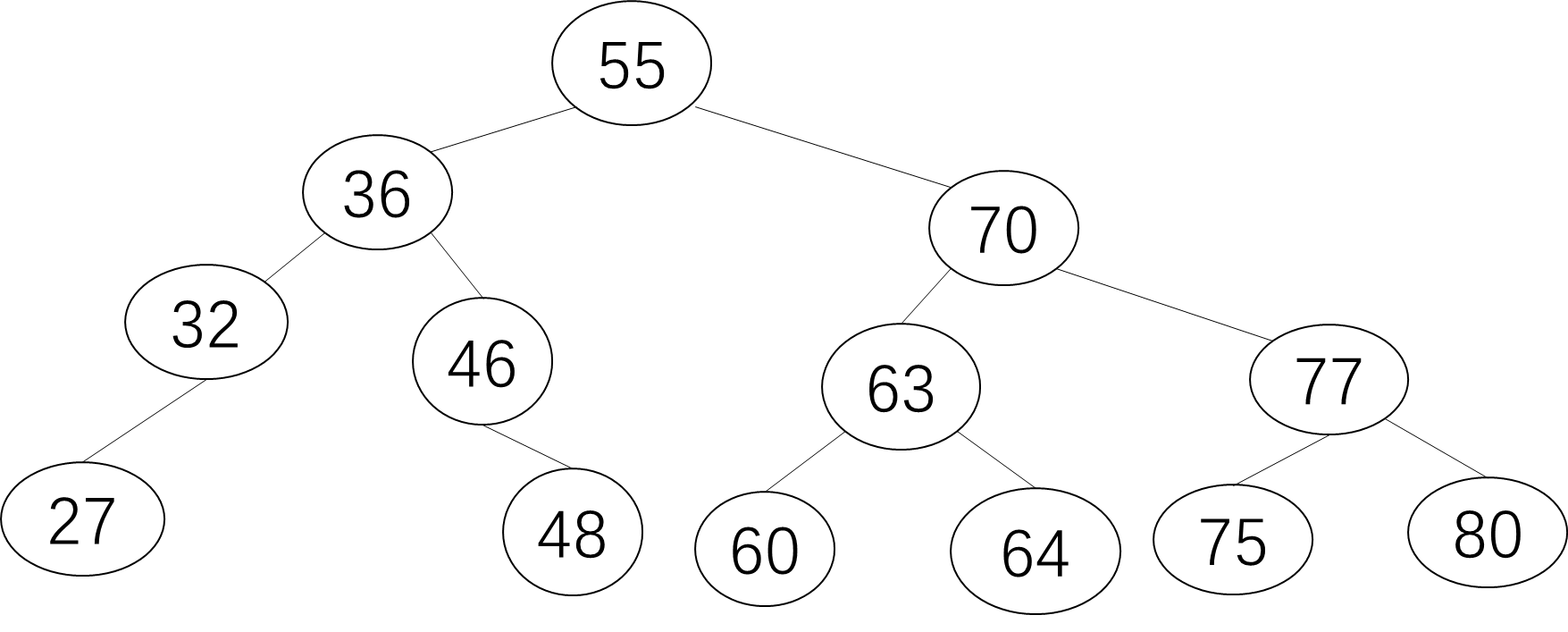
第十四趟：27 32 36 46 48 53 55 60 63 64 70 75 77 80

（2）答：下划线表示当前比较的元素

查找第一次: 27 32 36 46 48 53 55 60 63 64 70 75 77 80 32小于55，在序列左边

查找第二次: 27 32 36 46 48 53 55 60 63 64 70 75 77 80 32小于36，在序列左边

查找第三次: 27 32 36 46 48 53 55 60 63 64 70 75 77 80 32等于32，查找到数组的下标为1

（3）

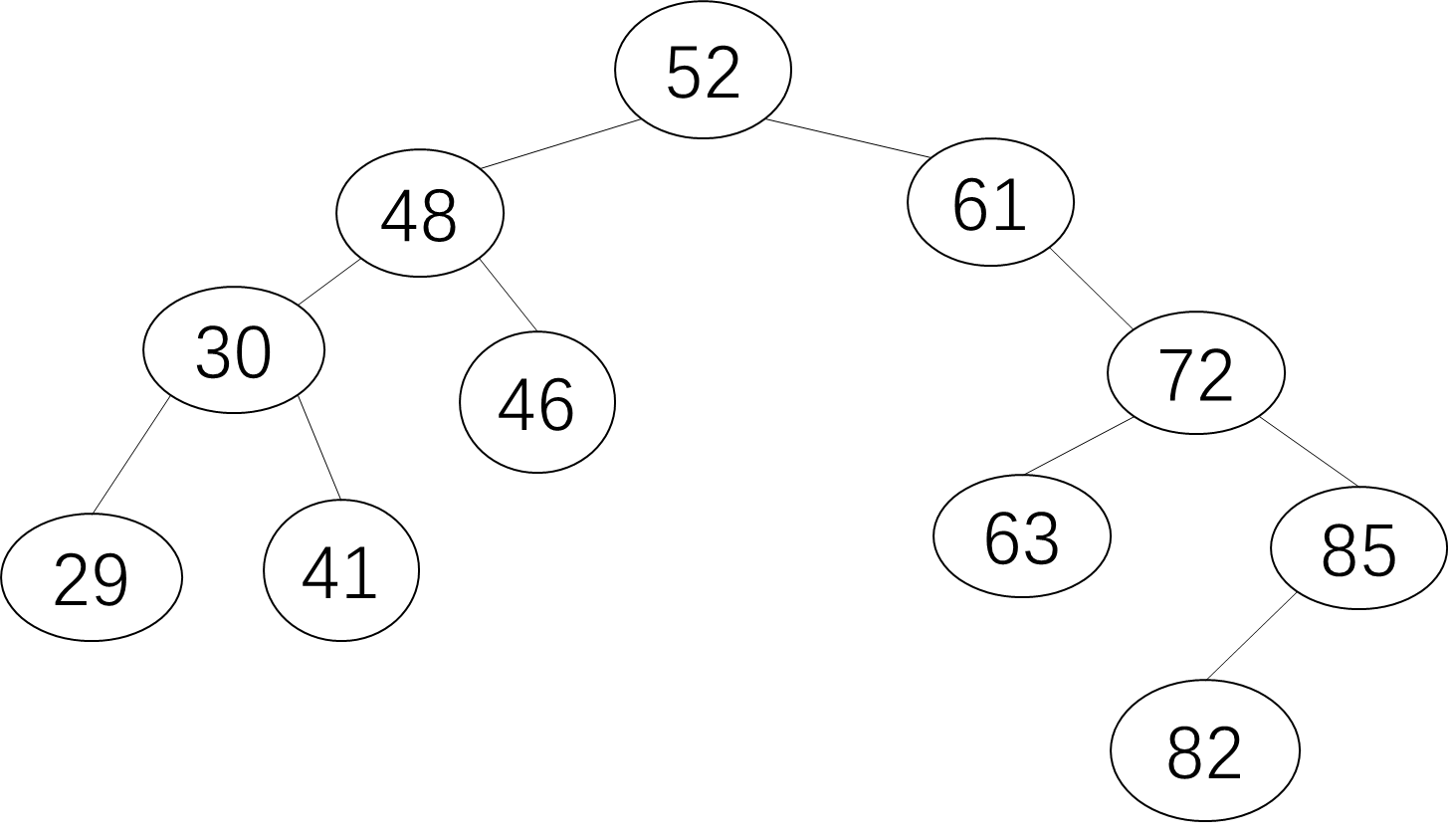
4、已知一组序列{52，48，30，61，29，41，72，63，85，82 }，

（1）根据序列构建二叉排序树,并计算这个二叉排序树查找成功的平均比较次数。

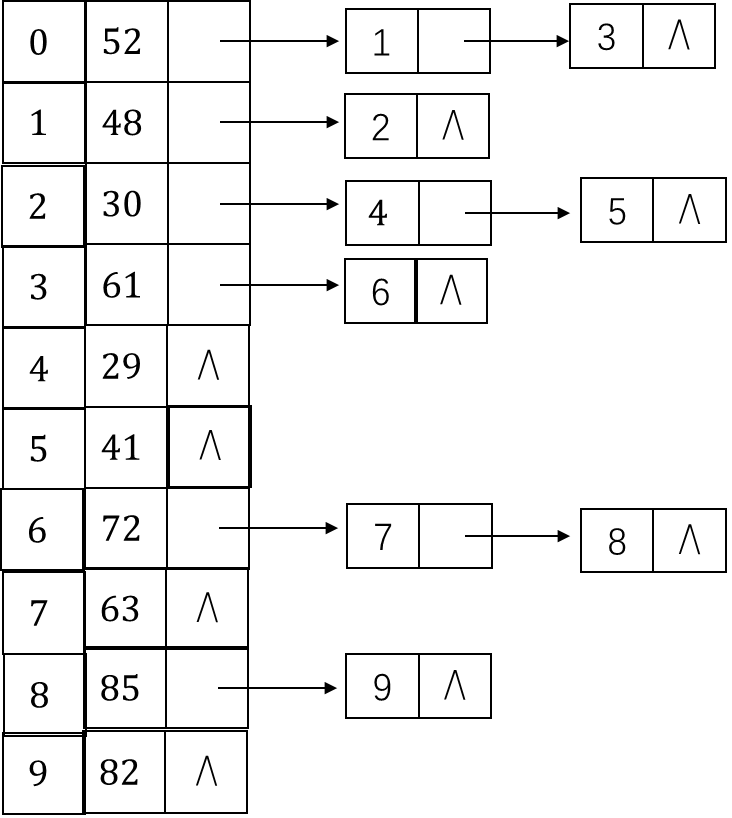
（2）根据（1）中的二叉排序树，采用孩子表示法表示这棵树。

（3）请写出二叉排序树的中序和后序遍历。

（1）答：

平均比较次数：(1+2\*2+3\*3+4\*4+5\*1)/11=3.18

(2)



(3)中序遍历：29 30 41 48 52 61 63 72 82 85

后序遍历: 29 41 30 48 63 82 85 72 61 52

5、已知散列函数H(k) = k mod 12，关键码集合{25,37,52,43,84,99,120,15,26,11,70,82},采用线性探测法处理冲突，试构造散列表（表长为12），并计算查找成功的平均查找长度。

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 84 | 25 | 37 | 99 | 52 | 120 | 15 | 43 | 26 | 82 | 70 | 11 |

查找成功的平均长度：（1+2+1+1+1+1+6+4+7+1+1+12）/12=38/12

6、已知一颗二叉树的先序、中序、后序遍历分别为：

先序遍历：□BC□E□GH

中序遍历：C□DA□GHF

后序遍历：□DB□□FEA

1. 其中有些字母已模糊不清，用□替代，请填补模糊不清的字母，给出完整的先序、中序、后序遍历序列。
2. 画出这颗二叉树。
3. 将该二叉树转成森林，画出该森林。

答：（1）

先序遍历：ABCDEFGH

中序遍历：CBDAEGHF

后序遍历：CDBHGFEA

(2)

(3)

7、已知记录序列{70、83、100、65、10、32、7、65、9}。写出用快速排序法进行升序排序的过程（写出每趟排序的结果）以及最终排序结果。

答：表示方式可以多样化，但是一定要清楚表述出排序的过程。

第一趟排序结果： {9 65 7 65 10 32} 70 {100 83}

第二趟排序结果： {7} 9 {65 65 10 32} 70 {100 83}

第三趟排序结果： 7 9 {65 65 10 32} 70 {100 83}

第四趟排序结果： 7 9 {32 65 10} 65 70 {100 83}

第五趟排序结果： 7 9 {10} 32 {65} 65 70 {100 83}

第六趟排序结果： 7 9 10 32 {65} 65 70 {100 83}

第七趟排序结果： 7 9 10 32 65 65 70 {100 83}

第八趟排序结果： 7 9 10 32 65 65 70 {83} 100

第九趟排序结果： 7 9 10 32 65 65 70 83 100

二、编程题

1、用C++语言编写一个两个一元多项式相加的程序（写出必要注释）。一元多项式存放在单链表中，单链表存储结点如图2所示：其中coef是非零项的系数，exp是指数，链表的节点按照指数大小升序排序。

coef

exp

next

图2

提示：对于一个一元多项式，它的单链表表示成图3。

4

0

7

2

8

5

∧

head

图3

答：详见课本62页，

1. 给定一个单链表，编写类函数SortList(),该函数采用直接插入排序将单链表的元素升序排序。

void LinkList<DataType>::SortList()

{

Node<DataType> \*p = first->next;

Node<DataType> \*q = NULL;

Node<DataType> \*s = NULL;

first->next = NULL;

while(p)

{

cout<<p<<endl;

q = first;

while(q->next!=p)

{

if((q->next==NULL)||(q->next->data>p->data))

{

s = p->next;

p->next = q->next;

q->next = p;

p = s;

break;

}

else

{

q = q->next;

}

}

if(p==q->next)

{

p = p->next;

}

}

}

3、编写func1函数，使得单链表()变为() 。即对单链表后半部分数据进行倒置。

void LinkList::func1()

{

Node\*p = first->next;

Node \*q = NULL;

Node \*s = NULL;

if(p)

{

q = p->next;

}

while(q&&q->next!=NULL)

{

p = p->next;

q = q->next->next;

}

q = p->next;

p->next = NULL;

while(q)

{

s = q->next;

q->next = p->next;

p->next = q;

q = s;

}

}