

3卷

一、单选题：

C D C C B A C A B D

二、填空题：

curr->next==head

63

71

5

31

$n(n+1)/2$

$(i-1)*k + m + 1$

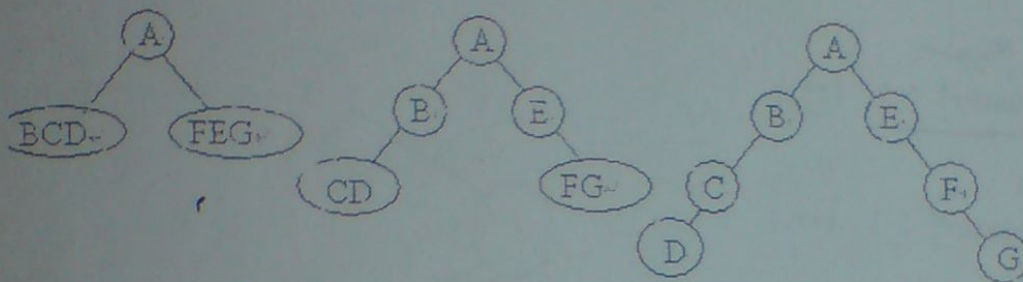
10,15,12

快速排序

shell 排序

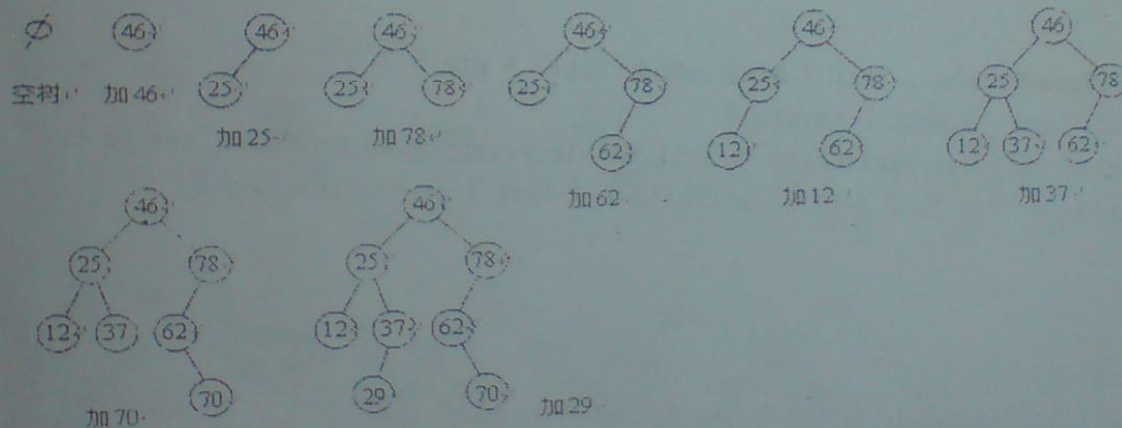
三、应用题

1、参考答案



先序遍历：ABCDEFG

2、

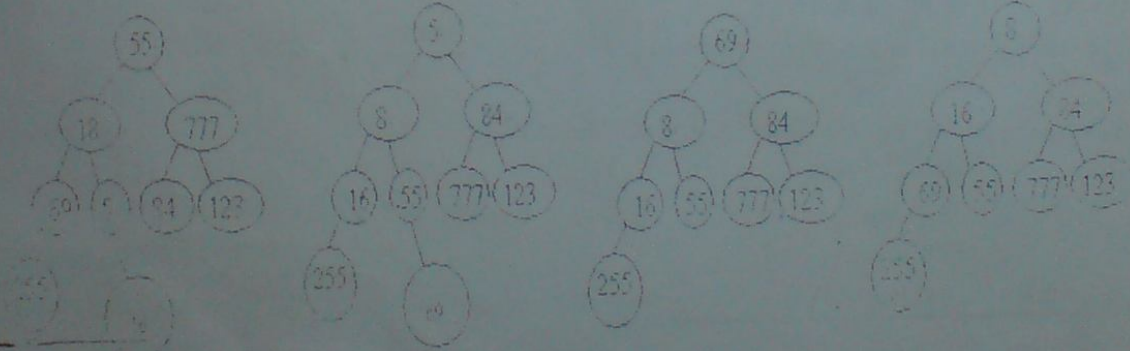


3、 初始

最小堆

第一个堆顶元素出堆

调整后的堆



- | | |
|---------------------------------------------------------------------------|---|
| $A \rightarrow C$ | 3 |
| $A \rightarrow B$ | 5 |
| $A \rightarrow B \rightarrow G$ | 6 |
| $A \rightarrow B \rightarrow G \rightarrow E$ | 7 |
| $A \rightarrow B \rightarrow G \rightarrow E \rightarrow F$ | 8 |
| $A \rightarrow B \rightarrow G \rightarrow E \rightarrow F \rightarrow D$ | 9 |

5.

0	1	2	3	4	5	6	7	8				
58	10	100	3	200	32	400	0		45	126	29	
(1)	(1)	(2)	(1)	(3)	(1)	(3)	(8)		(1)	(2)	(1)	

搜索成功的平均搜索长度为

$$ASL_{succ} = \frac{1}{11} (1 + 1 + 2 + 1 + 3 + 1 + 3 + 8 + 1 + 2 + 1) = \frac{24}{11}$$

四、算法设计题

```
1、 sort (int *a, int n)
```

```

{ int i, j; i=0, j=n-1;
  while (i<=j)
  {   if (a[i] % 2 == 1)   i++;
      if (a[j] % 2 == 0)   j--;
      if (i<j) { t=a[i]; a[i++]=a[j]; a[j--]=t; }
  }
  i=0;
  while (i<=j)
  {   if (prime(j))   { t=a[i]; a[i++]=a[j]; a[j--]=t; }
  }
  int prime(int k)
  {   int i;
      for (i=2; i<=sqrt(k); i++)
      if (k%i==0) return 0;
      return(1);
  }
}

```

```
2. void BinaryTree<Type> :: exchange ( BinTreeNode<Type> * ptr ) {  
    BinTreeNode<Type> * temp;  
    if ( ptr->leftChild != NULL || ptr->rightChild != NULL ) {  
        temp = ptr->leftChild;  
ptr->leftChild = ptr->rightChild;  
        ptr->rightChild = temp;  
        exchange ( ptr->leftChild );  
exchange ( ptr->rightChild );  
    }  
}
```


C

试卷一答案

一、选择题

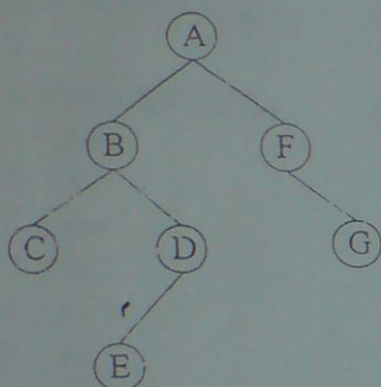
1~5 题	C	D	B	A	D
6~10 题	C	A	B	B	B

二、填空题。

1. $O(n^2)$ 2. $n-i+1$ 3. ABD//EG//C/F// 4. $2n-1$ 5. n
 6. abcefd 7. 0 8. $n-1$ 9. 1 10. 减少 1

三、应用题。

1.



2.

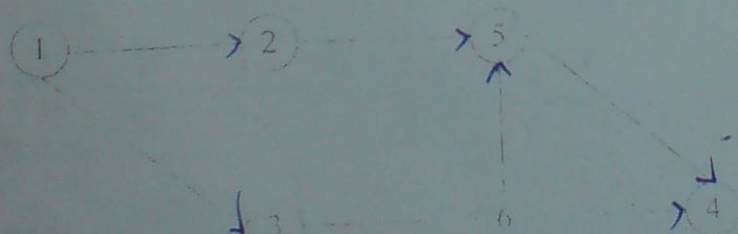
1	38	19	65	13	49	41	73
1	13	38	19	65	41	49	73
1	13	19	38	41	65	49	73
1	13	19	38	41	49	65	73
1	13	19	38	41	49	65	73
1	13	19	38	41	49	65	73
1	13	19	38	41	49	65	73

3.

散列表如下:

0	1	2	3	4	5	6
14	18	23	9	30	12	6

4.



拓扑序列:

123654 132654 136254

4.

用Prim算法求最小生成树:

(1,2)3 (1,3)5 (1,4)8 (4,6)4 (2,5)10 (4,7)20

四. 算法设计

1.

```
int CountX(LNode* L, ElemType x)
{   int i=0; LNode* p=L; //i 为计数器
    while(p!=NULL)
    {   if (P->data==x) i++;
        p=p->next;
    } //while, 出循环时 i 中的值即为 x 结点个数
    return i;
} //CountX
```

2.

```
(low<=high)      K==A[mid].key      Binsch(A, mid+1, hight, K)      return -1
```

3.

```
typedef struct node {int data; struct node *lchild,*rchild;} bitree;
void swapbitree(bitree *bt)
{
    bitree *p;
    if(bt==0) return;
    swapbitree(bt->lchild); swapbitree(bt->rchild);
    p=bt->lchild; bt->lchild=bt->rchild; bt->rchild=p;
}
```


一、选择题。

1~5 题	B	C	D	B	A
6~10 题	C	A	D	D	B

二、填空题。

1. $O(n)$ 2. n 3. $p \rightarrow lchild == NULL \ \&\& \ p \rightarrow rchild == NULL$ 4. 4 5. $2n-1$
6. $N-1$ 7. $2e$ 8. 0 9. 1 10. 减少 1

三、应用题。

1.

后序序列为 DEBCA

2.

Huffman 编码如下:

A 1110

B 1111

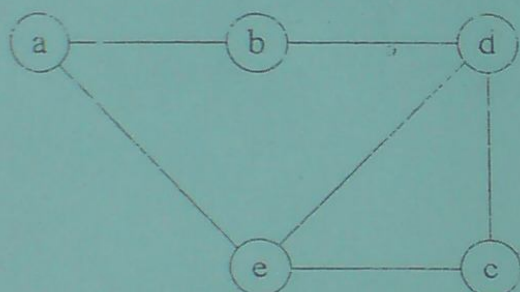
C 110

D 00

E 01

F 10

3.



深度优先序列: abdce

广度优先序列: abedc

4.

用克鲁斯卡尔算法得到的最小生成树为:

(1,2)3, (4,6)4, (1,3)5, (1,4)8, (2,5)10, (4,7)20

5.

散列表如下:

0	1	2	3	4	5	6
14	18	23	9	0	12	6

四、算法题。

1.

先序遍历二叉树, 打印每个结点的值以及该结点的层次数。

2.

BST \rightarrow data left right false

3.

```

int Eriseik(hlist *head)
{
    if(head == NULL || head->next == NULL)
    
```