

广东工业大学

2018 年攻读硕士学位研究生入学考试试题

考试科目(代码)名称: (829) 数据结构

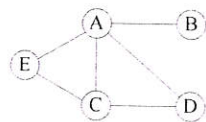
满分 150

(考生注意: 答卷封面需填写自己的准考证编号, 答完后连同本试题一并交回!)

一. 选择题(共 30 分, 15 小题, 每题 2 分)

- 对于长度为 n 的单链表, 算法时间复杂度是 $O(1)$ 的操作是 ()。
A. 查找操作
B. 删除 p 结点的后继结点
C. 在第 i 位置插入结点
D. 访问第 i 位置元素
- 单链表的优点是 ()。
A. 元素的存储空间是连续的
B. 插入删除不需要移动元素
C. 查找元素的效率高
D. 随机存取元素效率高
- 假设一个栈的进栈顺序为 1, 2, 3, 4, 则不可能得到的出栈序列是 ()。
A. 1 2 4 3 B. 4 1 3 2 C. 2 3 4 1 D. 3 2 4 1
- 若在循环队列中用数组 $base[0..m-1]$ 存储队列元素, 队头位置和队尾位置分别为 $front$ 和 $rear$, 则队列长度为 ()。
A. $(rear-front+m)\%m$
B. $rear-front$
C. $(rear-front+1)\%m$
D. $(rear-front-1)\%m$
- 已知无头结点的链队列的长度大于 1, p 和 q 分别为头指针和尾指针, 则出队的核心操作是 ()。
A. $q = q->next;$
B. $p->next = p;$
C. $q->next = q;$
D. $p = p->next$
- 若某完全二叉树按层遍历的序列为 ABCDEFGHI, 则该完全二叉树的先序遍历序列为 ()。
A. ABDHIECFG
B. HDBIAEFCG
C. HDBAIEFCG
D. ABDHEICFG
- 对一棵具有 n 个结点的二叉树, 若采用二叉链表存储结构, 则空指针域的个数是 ()。
A. n B. $n-1$ C. $n+1$ D. 以上答案均不对
- 在一棵高度为 h 的平衡二叉树中, 若每个结点的平衡因子均为 0, 则该树的结点个数为 ()。
A. h B. 2^h C. 2^{h-1} D. 2^h-1

9. 具有 n 个顶点的连通图的边数至少为 ()。
- A. $n+1$ B. n C. $n-1$ D. $n-2$
10. 具有 n 个顶点和 e 条边的无向图, 若采用邻接矩阵表示, 则矩阵中非零元素个数为 ()。
- A. n^2 B. e C. n^2-2e D. $2e$
11. 如图所示的无向图 G 的顶点 D 的度为 ()。
- A. 3 B. 2 C. 1 D. 0
12. 对长度为 8 的有序表进行折半查找, 在等概率情况下查找成功的平均查找长度为 ()。
- A. $21/8$ B. $24/8$ C. $19/8$ D. $25/8$
13. 一棵 m 阶 B 树中, 非根结点所包含的关键字的个数最多为 ()。
- A. $\lceil m/2 \rceil - 1$ B. m C. $m+1$ D. $m-1$
14. 不属于内部排序方法的是 ()。
- A. 希尔排序 B. 拓扑排序 C. 归并排序 D. 堆排序
15. 下列排序算法中, 不稳定的排序算法的是 ()。
- A. 冒泡排序 B. 直接插入排序 C. 基数排序 D. 快速排序

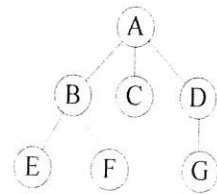


二. 填空题 (共 20 分, 10 小题, 每题 2 分)

1. 链式映像的特点是借助_____来表示数据元素之间的逻辑关系。
2. 对于不带头结点的单循环链表 L , 判断其为空表的条件是_____。
3. 在长度为 n 的顺序表中删除一个元素, 等概率情况下移动元素次数的期望值是_____。
4. 对初始为空的栈 s 执行一组操作 $\text{push}(s, 5); \text{push}(s, 3); \text{push}(s, 7); \text{pop}(s, x); \text{pop}(s, x);$ 后, x 的值为_____。
5. 二叉树中结点的子树根称为该结点的_____。
6. 若一棵哈夫曼树有 5 个叶子结点, 其权值分别为 $(1, 2, 4, 6, 8)$, 则其带权路径长度是_____。
7. 弗洛伊德算法用来求解_____的最短路径。
8. 对具有 n 个顶点和 e 条边的连通带权图, 利用普里姆算法构造最小生成树的时间复杂度为_____。
9. 将关键字序列 $(5, 6, 4, 2, 3)$ 的元素依次插入而构成的二叉排序树的深度是_____。
10. 简单选择排序在最坏情况下需要交换元素的次数为_____次。

三. 解答题 (共 42 分, 6 小题, 每题 7 分)

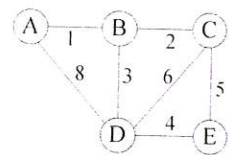
1. 对如图所示的树 T 构造其孩子链表表示。



2. 已知某二叉树的层序遍历序列为 ABCDEFGH, 中序遍历序列为 BFDGACHE, 请画出此二叉树。

3. 已知无向带权图 G 如图所示, 请:

- (1) 给出图 G 的邻接表;
- (2) 根据 (1) 中的存储结构, 给出从顶点 A 出发的广度优先遍历序列。

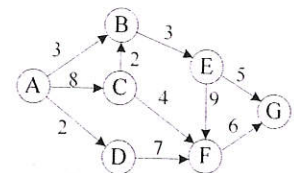


4. 已知 AOE 网如图所示, 其中各顶点的最早开始时间 ve 和最迟开始时间 vl 如下所示, 请:

- (1) 计算各活动弧的最早开始时间 e 和最迟开始时间 l ;
- (2) 列出关键路径。

	A	B	C	D	E	F	G
ve	0	10	8	2	13	22	28
vl	0	10	8	14	13	22	28

	AB	AC	AD	CB	BE	EF	CF	DE	EG	FG
e										
l										



5. 已知关键字序列 (12, 47, 69, 57, 14, 36, 78, 52), 哈希表地址区间长度为 11, 哈希函数为 $h(key) = (key * 3) \% 11$, 采用二次探测法解决冲突. 请构造哈希表, 并求等概率情况下查找成功时的平均查找长度。

6. 已知关键字序列 (7, 5, 3, 6, 2, 1, 8, 9), 将其调整为大顶堆。

四. 算法阅读题 (共 24 分, 3 小题, 每题 8 分)

1. (8 分) 已知带头结点的单链表 L, 阅读算法 f1, 回答下列问题:

(1) (4 分) 若 $L = (12, 4, 8, 17, 21, 10)$, 且 $e = 13$, 请写出执行算法 f1(L, e) 后的 L;

(2) (4 分) 简述算法 f1 的功能。

```
void f1(LinkList &L, ElemType e) {
    if(L->next!=NULL) {
        if(L->next->data < e) {
            p = L->next;  L->next = p->next;
            free(p);  f1(L, e);
        }
        else f1(L->next, e);
    }
}
```

2. (8 分) 已知非空栈 S, 阅读算法 f2, 回答下列问题:

(1) (4 分) 设栈 $S = (1, 2, 3, 4, 5, 6)$, 请写出执行算法 f2(S, e) 后的 S 和 e 的值;

(2) (4 分) 简述算法 f2 的功能。

```
void f2(Stack &S, ElemType &e) {
    InitStack(S1);
    while(!StackEmpty(S)) {
        Pop(S, e1);
        Push(S1, e1);
    }
    Pop(S1, e);
    while(!StackEmpty(S1)) {
        Pop(S1, e1);
        Push(S, e1);
    }
}
```

3. (8 分) 阅读算法 f3, 回答下列问题:

(1) (4 分) 若顺序表 $L = (12, 50, 24, 76, 63)$, 请写出执行算法 f3(L) 后的 L;

(2) (4 分) 简述算法 f3 的功能。

```
void f3(SqList &L) {
    for(i = 0; i<L.length-1; i++) {
        m = L.elem[i]; j = i;
```

```

for(k = i+1; k<L.length; k++)
    if(L.elem[k]<m) { m = L.elem[k]; j = k; }
if(i!=j) {
    temp = L.elem[i]; L.elem[i] = L.elem[j];
    L.elem[j] = temp;
}
}
}

```

五. 算法填空题 (共 24 分, 3 小题, 每题 8 分)

1. (8分) 设双向链表的类型定义如下:

```

typedef struct DuLNode {
    ElemType data;
    struct DuLNode *prior, *next;
} DuLNode, *DuLinkList;

```

算法f4在带头结点的双向链表L中将p指针所指结点与其直接前驱结点进行交换, 若交换成功, 返回OK, 否则, 返回ERROR。请在空缺处填入合适内容, 使其成为完整的算法。

```

Status f4(DuLinkList &L, DuLinkList p) {
    q = ①;
    if(q==L) return ERROR;
    q->next = p->next;
    if(p->next!=NULL) ②;
    q->prior->next = p;
    ③;
    p->next = q;
    ④;
    return OK;
}

```

2. (8分) 已知树T采用孩子兄弟存储结构, 其类型定义如下:

```

typedef struct CSTNode {
    TElemType data;
    struct CSTNode *firstChild, *nextSibling;
} CSTNode, *CSTree;

```

算法f5(T, e, 1)在树T中查找结点e, 若存在, 则返回其层次, 否则返回0。请在空缺处填入合适内容, 使其成为完整的算法。

```

int f5(CSTree T, TElemType e, int lev) {
    if( ① ) return 0;
    if( ② ) return lev;
    res = f5( ③ , e, lev+1);
    if(res > 0) return res;
    return ④ ;
}

```

3. (8分) 设无权图 G 采用数组表示法存储结构, 其类型定义如下:

```

#define MaxNum 5
typedef struct {
    char vexs[MaxNum]; // 顶点表
    int arcs[MaxNum][MaxNum]; // 邻接矩阵
    int n, e; // 结点数和边数
} MGraph;

```

算法 f6 在图 G 中求顶点 v 相对于顶点 w 的下一个邻接顶点, 若存在, 返回顶点值, 否则返回 '#'。请在空缺处填入合适内容, 使其成为完整的算法。

```

char f6(MGraph G, char v, char w) {
    for(i = 0; i < G.n && G.vexs[i] != v; i++);
    for(j = 0; j < G.n && ① ; j++);
    if(i >= G.n || j >= G.n || 0 == G.arcs[i][j])
        return '#';
    for( ② ; k < G.n; k++)
        if( ③ != 0) return ④ ;
    return '#';
}

```

六. 算法设计题 (共 10 分, 1 题)

设二叉树采用二叉链表存储结构, 其类型定义如下。试编写递归算法在二叉树中查找值为 e 的一个结点, 若存在, 则打印出其所有的祖先结点。

```

typedef struct BiTNode {
    char data;
    struct BiTNode *lchild, *rchild;
} BiTNode, *BiTree;

```