

2025 年 4 月高等教育自学考试
数据结构与算法试题
课程代码:13003

1. 请考生按规定用笔将所有试题的答案涂、写在答题纸上。
2. 答题前,考生务必将自己的考试课程名称、姓名、准考证号用黑色字迹的签字笔或钢笔填写在答题纸规定的位置上。

选择题部分

注意事项:

每小题选出答案后,用 2B 铅笔把答题纸上对应题目的答案标号涂黑。如需改动,用橡皮擦干净后,再选涂其他答案标号。不能答在试题卷上。

一、单项选择题:本大题共 15 小题,每小题 2 分,共 30 分。在每小题列出的备选项中只有一项是最符合题目要求的,请将其选出。

1. 下列关于数据的逻辑结构的叙述中,不正确的是
- A. 数据的逻辑结构是数据间关系的描述
 - B. 数据的逻辑结构抽象反映数据元素间的逻辑关系
 - C. 数据的逻辑结构具体反映数据在计算机中的存储方式
 - D. 数据的逻辑结构分为线性结构和非线性结构
2. 算法分析要评估的两个主要方面是
- A. 空间复杂度和时间复杂度
 - B. 正确性和简明性
 - C. 可读性和文档性
 - D. 数据复杂性和程序复杂性
3. 将斐波那契数列前 n 项保存在数组中,设计算法时适宜使用的策略是
- A. 分治法
 - B. 穷举法
 - C. 递归法
 - D. 递推法
4. 设顺序表中有 n 个数据元素,若删除表中第 i 个元素则需要移动的元素个数是
- A. i
 - B. $n+1-i$
 - C. $n-1-i$
 - D. $n-i$



5. 在头指针为 head 的非空单向循环链表中, 指针 p 指向尾结点, 下列关系成立的是
A. $p \rightarrow next == head$
B. $p \rightarrow next \rightarrow next == head$
C. $p \rightarrow next == NULL$
D. $p == head$
6. 一个栈的输入序列为 1, 2, 3, ..., n, 若输出序列的第一个元素是 n, 则第 i ($1 < i \leq n$) 个输出元素是
A. $n-i-1$
B. $n-i+1$
C. $n-i$
D. i
7. 入栈序列是 1, 2, 3, 4, 出栈序列是 2, 4, 3, 1, 则栈的容量最小是
A. 1
B. 2
C. 3
D. 4
8. 已知循环队列存储在一维数组 A[0...n-1] 中, 且队列非空时 front 和 rear 分别指向队头元素与队尾元素。若初始时队列为空, 且要求第一个进入队列的元素存储在 A[0] 处, 则初始时 front 和 rear 的值分别是
A. 0, 0
B. 0, n-1
C. n-1, 0
D. n-1, n-1
9. 广义表 ((a), a) 的表头和表尾分别是
A. a, ((a))
B. ((a)), a
C. (a), a
D. (a), (a)
10. 设有两个串 p 和 q, 其中 q 是 p 的子串, 求 q 在 p 中首次出现的位置的运算称为
A. 模式匹配
B. 联接
C. 求子串
D. 求串长
11. 若一棵二叉树中度为 1 的结点个数是 3, 度为 2 的结点个数是 4, 则该二叉树叶子结点的个数是
A. 4
B. 5
C. 7
D. 8
12. 求带权图单源最短路径的算法称为
A. 迪杰斯特拉 (Dijkstra) 算法
B. 克鲁斯卡尔 (Kruskal) 算法
C. 普里姆 (Prim) 算法
D. 广度优先搜索算法
13. 平均时间复杂度为 $O(n \log n)$ 的稳定排序算法是
A. 快速排序
B. 堆排序
C. 归并排序
D. 冒泡排序

14. 分块查找方法将表分为多块，并要求

- A. 块内有序
- B. 各块等长
- C. 块间有序
- D. 链式存储

15. 下列关键字序列中，构成大根堆的是

- A. 5, 8, 1, 3, 9, 6, 2, 7
- B. 9, 8, 1, 7, 5, 6, 2, 3
- C. 9, 8, 6, 3, 5, 1, 2, 7
- D. 9, 8, 6, 7, 5, 1, 2, 3

非选择题部分

注意事项：

用黑色字迹的签字笔或钢笔将答案写在答题纸上,不能答在试题卷上。

二、填空题：本大题共 10 小题，每小题 2 分，共 20 分。

- 16. 在数据结构中，_____作为一个完整的对象（整体）是构成数据的基本单位。
- 17. 算法中要做的运算都是相当基本的、能够精确进行的，这个特性是算法的_____。
- 18. 线性表基本操作的具体实现需要依赖线性表的_____。
- 19. 在有 n 个顶点的无向图中，其边数最多可达_____。
- 20. 高度为 4 的 AVL 树中，最少有_____个结点。
- 21. 在有序表 (1 3 9 12 32 41 45 62 75 77 82 95 100) 中，进行折半查找时，若查找关键字 9 时的比较次数为 2，则查找关键字 75 时的比较次数是_____。
- 22. 影响排序效率的两个因素是关键字的_____次数和记录的移动次数。
- 23. 对特殊矩阵进行压缩的目的是节省_____。
- 24. 对序列 {55, 46, 13, 94, 17, 42} 进行基数排序，第一趟排序后的结果是_____。
- 25. 设某棵二叉树的中序遍历序列为 ABCD，后序遍历序列为 BADC，则其先序遍历序列为_____。

三、解答题：本大题共 4 小题，每小题 5 分，共 20 分。

- 26. 列出五种构造哈希函数的方法。
- 27. 含有 n 个顶点的连通图 G 的最小生成树有哪些性质？
- 28. 设二维数组 $A[5][6]$ 的每个元素占 4 个字节，已知数组首地址为 1000， A 共占多少个字节？分别按行和列优先存储时， $A[2][5]$ 的起始地址分别为多少？
- 29. 简述二叉树先序遍历算法的过程。



四、算法阅读题：本大题共 4 小题，每小题 5 分，共 20 分。

30. 阅读下列程序，并回答问题：

```
LinkedList mynote(LinkedList L)    //L 是不带头结点的单链表的头指针
{
    if(L&&L->next)
    {
        q=L; L=L->next; p=L;
S1:    while(p->next) p=p->next;
S2:    p->next=q; q->next=NULL;
    }
    return L;
}
```

(1) 说明语句 S1 及 S2 的功能；

(2) 设链表表示的线性表为 $(a_1, a_2, \dots, a_n)$ ，写出算法执行后的返回值所表示的线性表。

31. 阅读下列程序，并回答问题：

```
#include<stdio. h>

substr(char *t, char *s, int pos, int len)
{
    while(len>0&&*s)
    {
        *t=* (s+pos-1);
        t++;s++;len--;
    }
    *t=' \0' ;
}

char *func(char *s)
{
    char t[100];
    if (strlen(s)=1)    return s;
    substr(t, s, 1, 1); substr(s, s, 2, strlen(s)-1);
    func(s);
    return strcat(s, t);
}

main( )
{
    char str[100]="String";
    printf("%s\n", func(str));
}
```

(1) 请写出执行该程序后的输出结果；

(2) 简述函数 func 的功能。



32. 阅读下列算法，并回答问题：

（注：InitQueue、EnQueue、DeQueue 和 QueueEmpty 分别是队列初始化、入队、出队和判队空的操作）

```
void func(Queue *Q, Queue *Q1, Queue *Q2)
{ int e;
  InitQueue(Q1); InitQueue(Q2);
  while(!QueueEmpty(Q))
  { e=DeQueue(Q);
    if(e>=0) EnQueue(Q1, e);
    else EnQueue(Q2, e)
  }
}
```

(1) Q、Q1 和 Q2 都是队列结构，设队列 Q= (2, 0, -8, 5, -3, -7, 6)，其中 2 为队头元素，写出执行 func(&Q, &Q1, &Q2) 之后队列 Q、Q1 和 Q2 的状态。

(2) 简述算法 func 的功能。

33. 下面程序实现插入排序算法，请在空白处填上适当内容以将算法补充完整。

```
typedef struct {int key; Info otherinfo;} SeqList;
void InsertSort(SeqList R[], int n)
{ SeqList x;
  int i, j, k, lo, hi, mi;
  for (i=2; i<=n; i++)
  { __①__ ; lo=1; hi=i-1;
    while (lo<=hi)
    { mi=(lo+hi)/2;
      if (__②__) break;
      if (R[mi].key>x.key) hi=mi-1;
      else lo=__③__;
    }
    if (mi==lo) k=__④__;
    else k=i-mi-1;
    for (j=0; j<k; j++)
      __⑤__;
    R[i-j]=x;
  }
}
```



五、算法设计题：10 分。

34. 无向图 G 采用邻接矩阵存储，存储定义如下：

```
define MAXV 50//最大顶点个数  
typedef struct{  
    int arcs[MAXV][MAXV]; //邻接矩阵，顶点关系（0 或 1）  
    int vexnum, arcnum; //顶点数，弧数  
    VertexType vxs[MAXV]; //存放顶点信息  
} MGraph;
```

设计算法求图 G 中顶点度的最大值。

- (1) 给出算法的基本设计思想。（4 分）
- (2) 根据设计思想，采用 C 或 C++ 语言描述算法，关键之处给出注释。（6 分）

