

华东高校计算机基础教育研究会推荐教材

数据结构习题解析 与上机实验指导

宁正元 易金聪 等编著



中国水利水电出版社
上海交通大学出版社
东南大学出版社

华东高校计算机基础教育研究会推荐教材

数据结构习题解析与上机实验指导

宁正元 易金聪 等编著

中国水利水电出版社
上海交通大学出版社
东南大学出版社

内 容 简 介

本书是为了配合各类专业《数据结构》课程而编写的，与作者 2000 年 6 月出版的教材《数据结构——用 C 语言描述》相配套。除了给出该教材中习题的提示、解析和答案外，还针对各章内容适当地补充了练习题目，并给出答案或解析；针对教学内容组织了上机实习题目，并给予必要的上机指导；为了满足教学和各类考生学习课程与考前复习的需求，还给出了十套模拟试题及答案。

全书内容丰富、题型多样、涉及面广、实用性强，与《数据结构》课程主要内容紧密结合。可供各类学生课程学习与考前复习使用，也可供教师或其他专业技术人员参考。

图书在版编目(CIP)数据

数据结构习题解析与上机实验指导 / 宁正元 等编著. —北京：中国水利水电出版社，2000.9

华东高校计算机基础教育研究会推荐教材

ISBN 7-5084-0458-0

I. 数… II. 宁 … III. 数据结构-高等学校-教学参考资料 IV. TP311.12

中国版本图书馆 CIP 数据核字 (2000) 第 47809 号

书 名	数据结构习题解析与上机实验指导
作 者	宁正元 易金聪等编著
出版、发行	中国水利水电出版社（北京市三里河路 6 号 100044） 网址：www.waterpub.com.cn E-mail: sale@waterpub.com.cn 电话：(010) 63202266（总机）、68331835（发行部）
经 售	全国各地新华书店
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	787×1092 毫米 16 开本 18.25 印张 410 千字
版 次	2000 年 9 月第一版 2001 年 7 月北京第三次印刷
印 数	10001—13000 册
定 价	23.00 元

凡购买我社图书，如有缺页、倒页、脱页的，本社发行部负责调换

版权所有·侵权必究

前 言

数据结构是计算机科学与技术专业教学计划中的一门核心课程，是计算机科学与技术专业和其他有志从事计算机与技术工作的人员的一门重要的专业基础课程。计算机科学与技术学科各领域都要用到各种数据结构，而且要从从事计算机科学与技术工作，尤其是计算机应用领域的开发研制工作，必须具备良好的数据结构基础。

数据结构课程的教学要求是学会分析研究计算机加工的数据对象的特征，以便在实际应用中选择适当的数据结构、存储结构和相应的算法，初步掌握算法的时间与空间性能分析技巧，得到复杂程序设计的训练。

数据结构课程内容丰富、学习量大；隐含在各部分内容中的方法和技术多，在先修课程中介绍的专业性知识又不多；贯穿全书的动态链表存储结构和递归技术初学者往往不易掌握，算法设计具有动态性和抽象性等特点，看懂听明白与掌握会应用之间有相当大的一段距离。这些特点使得这门课程的学习难度很大，许多学生在解答习题尤其是算法设计题时，觉得无从下手，似乎教学内容与习题毫不相干。事实上解答习题所需的各种方法和技术都在教科书中，只不过其呈现形式多样化，需仔细体会才能掌握。

我们在认真总结近二十年教学经验和体会的基础上，结合新时期大学生的学习特点和要求，编写这本《数据结构习题解析与上机实验指导》，以希望通过一些典型习题的解析和具体的实现环节来给学生一些解题示范和启发，帮助学生更好地学习和掌握课程内容，理解和掌握算法设计所需的方法和技术，为整个专业学习打下良好的基础。

本书是《数据结构——用 C 语言描述》的配套教材。全书内容共分三个部分。第一部分习题解析，主要是对教科书中的习题给出提示、解析和答案。每一章中的习题都按基础知识和应用知识两大类给出。基础知识类题目是针对本章基本内容给出的复习巩固知识点的题目；应用知识类题目是针对本章知识点的应用型题目，侧重点在各种基本方法与技术的运用能力训练方面。所给题目都在原教科书的基础上有一定量的补充，以期扩大学生的知识掌握面，也能使采用其他同类教科书学习的学生使用，适应有志报考计算机类硕士研究生的学生考前复习使用的需要。第二部分上机实验指导，给出了七个上机实验，每个实验都给出了几道上机实验题目，每个实验题目都介绍了实验目的，主要采用的方法与技术和 C 语言实现程序。通过实验使学生了解并学会如何运用数据结构知识去解决现实世界的某些实际问题，具备设计较复杂程序的初步能力。第三部分模拟试题，主要是为了帮助学生在学完课程内容后自我检验学习效果。有易、较易、中、较难、难五种难度的题目分布于各套试题中，每套试题中有选择、填空、判断、应用、设计等各种题型的题目。试题覆盖了教科书中的大多数知识点，有助于对课程的系统复习。

本书的出发点是帮助学生学好数据结构这门课程，所以在使用的过程中需要注意以下几点：第一要与课程内容的学习同步，有利于进一步理解掌握各知识点和巩固课堂教学效果。第二算法设计具有不唯一性，对算法设计题目本书只给出一种或几种算法答案，要在学习、理解、领会的基础上自己动手设计算法程序，这样才会取得良好的效果。切忌照抄照搬，

否则会影响学习效果。第三要遵循循序渐进的原则，本书内容按知识结构分类，按由易到难、由典型到一般的次序排列，读者最好能按次序阅读，但算法讲解是独立构件，也可以有选择地阅读。第四要学会举一反三，触类旁通，课程内容的知识点是有限的，但运用这些知识点、运用所介绍的方法和技术解决的实际问题却是无限的，重在掌握基本原理、基本方法和基本技术，并学会灵活运用。第五要有 C 语言的基础。本书算法是用 C 语言描述，所以读者应先了解 C 语言的基本内容。

本书是受华东高校计算机教材编委会的委托编写的。本书的第一部分中的第一、二、五、八、九章及第三部分由宁正元执笔，第一部分中的第三、四、六、七、十章及第二部分由易金聪执笔，全书由宁正元负责统稿、修改、定稿，西北大学康宝生博士后主审。感谢福州大学计算机系陈国龙博士提供了有关资料以及《数据结构——用 C 语言描述》一书的作者张健、邓桂英、朱穗颖提供了部分章节的部分习题解析或实验指导资料，感谢华东高校计算机教材编委会和中国水利水电出版社、上海交通大学出版社、东南大学出版社对本书的出版给予的大力支持。

由于时间仓促和作者水平所限，本书一定还存在着许多问题，敬请广大读者批评指正。

作者

2000年4月

目 录

前言

第一部分 习题解析

第一章 绪论	1
1.1 基础知识	1
1.2 应用知识	3
第二章 线性表	7
2.1 基础知识	7
2.2 应用知识	10
第三章 栈和队列	27
3.1 基础知识	27
3.2 应用知识	29
第四章 串	40
4.1 基础知识	40
4.2 应用知识	40
第五章 数组与广义表	48
5.1 基础知识	48
5.2 应用知识	49
第六章 树	67
6.1 基础知识	67
6.2 应用知识	69
第七章 图	89
7.1 基础知识	89
7.2 应用知识	92
第八章 检索	107
8.1 基础知识	107
8.2 应用知识	110
第九章 排序	126
9.1 基础知识	126
9.2 应用知识	129
第十章 文件	143
10.1 基础知识	143
10.2 应用知识	144

第二部分 上机实验指导

☆ 预备知识	152
☆ 实验内容	157
上机实习题一 线性表	157
上机实习题二 栈和队列	168
上机实习题三 数组与广义表	185
上机实习题四 树	189
上机实习题五 图	196
上机实习题六 查找	204
上机实习题七 排序	211

第三部分 模拟试题

模拟试题一	218
模拟试题二	220
模拟试题三	222
模拟试题四	225
模拟试题五	228
模拟试题六	230
模拟试题七	232
模拟试题八	235
模拟试题九	239
模拟试题十	242
模拟试题参考答案	246
参考文献	283

第一部分 习题解析

第一章 绪 论

1.1 基础知识

一、填空题

1. 数据结构即数据的逻辑结构包括 ①, ②, ③ 和 ④ 四种类型, 树型结构和图型结构合称为 ⑤, 数据的存储结构即物理结构包括 ⑥, ⑦, ⑧ 和 ⑨ 等四种基本类型。

2. 数据结构是研究数据的 ① 和 ② 以及它们之间的相互关系, 并对这种结构定义相应的 ③, 设计出相应的 ④, 而确保经过这些运算后所得到的新结构是 ⑤ 结构类型。

3. 一个数据结构用二元组表示时, 它包括 ① 集合 K 和 K 上 ② 的集合 R 。

4. 一个算法应具有 ①, ②, ③, ④ 和 ⑤ 这五个特性。

5. 一个算法的时间复杂度是该算法包含的 ① 的多少, 它是一个算法运行时间的 ②, 一个算法的空间复杂性是指该算法在运行过程中临时占用的 ③ 的大小。

6. 一个算法的时间复杂度通常用它的 ① 形式表示, 当一个算法的时间复杂度与问题的规模 n 大小无关时, 则表示为 ②; 成正比时, 则表示为 ③; 成对数关系时, 则表示为 ④; 成平方时, 则表示为 ⑤。

7. ① 是描述客观事物的数、字符以及所有能输入到计算机且被计算机程序加工处理的符号集合。

② 是数据的基本单位, 有时一个 ② 由若干个 ③ 组成, 在这种情况下, 称 ② 为记录, ③ 是数据的最小单位, 而由记录组成的线性表为 ④。

被计算机加工的 ② 不是孤立无关的, 它们彼此之间一般存在着某种联系, 通常将 ② 间的这种联系称为 ⑤。

算法的计算量的大小称为计算的 ⑥。

二、选择题

1. 下面程序的时间复杂度为_____。

```
for(i=0; i<m; i++)  
    for(j=0; j<n; j++)  
        A[i][j]=i*j;
```

(A) $O(m^2)$ (B) $O(n^2)$ (C) $O(m \times n)$ (D) $O(m+n)$

2. 下面程序段的执行次数为_____。

```
for(i=0; i<n; i++)
```



```
for(j=n; j<=i; j- -)
```

```
state;
```

(A) $n(n+2)/2$ (B) $(n-1)(n+2)/2$ (C) $n(n+1)/2$ (D) $(n-1)(n+2)$

3. 下列程序的时间复杂度为_____。

```
i=0; s=0;
```

```
while(s<n)
```

```
{
```

```
    i++;
```

```
    s=s+i;
```

```
}
```

(A) $O(\sqrt{n})$ (B) $O(1)$ (C) $O(n)$ (D) $O(n^2)$

4. 下列程序的时间复杂度为_____。

```
for(i=0; i<m; i++)
```

```
    for(j=0; j<t; j++)
```

```
        c[i][j]=0;
```

```
for(i=0; i<m; i++)
```

```
    for(j=0; j<t; j++)
```

```
        for(k=0; k<n; k++)
```

```
            c[i][j]=c[i][j]+a[i][k]*b[k][j];
```

(A) $O(m \times n \times t)$ (B) $O(m+n+t)$ (C) $O(m+n \times t)$ (D) $O(m \times t+n)$

5. 下列程序段的时间复杂度为_____。

```
i=1; k=0; n=100;
```

```
do{
```

```
    k=k+10*i;
```

```
    i=i++;
```

```
}while(i!=n);
```

(A) $O(1)$ (B) $O(n)$ (C) $O(i)$ (D) $O(i \times n)$

6. 下列程序段的时间复杂度为_____。

```
x=n;            /*n>1*/
```

```
y=0;
```

```
while((x>= (y+1)*(y+1))
```

```
    y=y+1;
```

(A) $O(n)$ (B) $O(\sqrt{n})$ (C) $O(1)$ (D) $O(n^2)$

7. 下列程序段的时间复杂度为_____。

```
m=91; n=100;
```

```
while(n>0)
```

```
    if (m>0)
```

```
{ m=m-10;
```

```
    n=n-1;
```

```

    }
    else m=m+1;

```

(A) $O(n^2)$ (B) $O(n)$ (C) $O(\sqrt{n})$ (D) $O(m)$

解答:

一、填空题

- ① 线性结构 ② 树型结构 ③ 图型结构 ④ 集合 ⑤ 非线性结构 ⑥ 顺序
⑦ 链接 ⑧ 索引 ⑨ 散列
- ① 物理结构 ② 逻辑结构 ③ 运算 ④ 算法 ⑤ 原来的
- ① 数据元素 ② 二元关系
- ① 有穷性 ② 确定性 ③ 可行性 ④ 0或多个输入 ⑤ 1或多个输出
- ① 简单操作次数 ② 相对量度 ③ 存储空间
- ① 数量级 ② $O(1)$ ③ $O(n)$ ④ $O(\log_2 n)$ ⑤ $O(n^2)$
- ① 数据 ② 数据元素 ③ 数据项 ④ 文件 ⑤ 结构 ⑥ 复杂度

二、选择题

1. C 2. C 3. A 4. A 5. A 6. B 7. B

1.2 应用知识

1. 选择解决某种问题的最佳数据结构的标准是什么?

解答: 一般有两条标准:

- (1) 所需的存储空间量。
- (2) 算法所需要的时间。

而算法所需要的时间又包括以下几点:

- (1) 程序运行时所需要的数据总量;
- (2) 源程序进行编译所需要的时间;
- (3) 计算机执行每条指令所需要的时间;
- (4) 程序中的指令重复执行的次数, 而本条正是讨论算法中的重点内容。

2. 根据大 O 表示法的意义, 证明大 O 表示法的加法法则。

解答: 首先给出加法法则如下:

若两个程序段的时间复杂度为 $T_1(n)=O(f_1(n))$ 和 $T_2(n)=O(f_2(n))$ 。那么这两个程序段依次执行时的时间复杂度为:

$$T(n)=T_1(n)+T_2(n)=O(\max(f_1(n), f_2(n)))$$

上式可以证明如下:

由于

$$T_1(n)=O(f_1(n)), T_2(n)=O(f_2(n))$$

则存在常数 c 和 n_0 , 使得 $n \geq n_0$ 时, 有

$$T_1(n)+T_2(n) \leq c \times \max(f_1(n), f_2(n))$$

例如, $T_1(n)=O(n^2)$, $T_2(n)=O(n^3)$

当 $n_0=1$ 时, 对所有 $n \geq n_0$ 都有

$$n^3+n^2 \leq n^3+n^3=2n^3$$

当取 $c=2$ 时, 则有 $T(n) \leq c \times n^3$, 取 $T(n)=O(n^3)$,

所以有

$$T(n)=T_1(n)+T_2(n)=O(\max(f_1(n), f_2(n)))$$

成立。

3. 将数量级 $O(1)$, $O(n)$, $O(n^2)$, $O(n^3)$, $O(n \log_2 n)$, $O(\log_2 n)$, $O(2^n)$ 按增长率从小到大排列。

解答: 上面的几种类型的数量级中 $O(1)$ 为常量型, $O(n)$ 为线性型, $O(n^2)$ 为平方型, $O(n^3)$ 为立方型, $O(\log_2 n)$ 为对数型, $O(2^n)$ 为指数型, $O(n \log_2 n)$ 为线性对数型, 它们按增长率从小到大的顺序为: $O(1)$, $O(\log_2 n)$, $O(n)$, $O(n \log_2 n)$, $O(n^2)$, $O(n^3)$, $O(2^n)$ 。

4. 编出在 20 个数中查找出最大或最小数的程序。

解答:

```
#define    n    20;
search(A);
int A[ ];
{
    int i, j, max, min;
    for(i=0; i<n; i++)
        scanf("%d\n", A[i]);
    min=A[0];
    max=A[0];
    j=1;
    while(j<=n-1)
    {
        if (A[j]>max)    max=A[j];
            if(A[j]<min)    min=A[j];
                j++;
    }
    printf("%d%d\n", max, min);
} /*search*/
```

5. 已知输入 x , y , z 三个不相等的整数, 试设计一个算法, 使这三个数按从小到大的顺序输出, 并考虑此算法的比较次数和元素的移动的次數。

解答:

```
sort(x, y, z);
int x, y, z;
{scanf("%d%d%d\n", &x, &y, &z);
```

```
    if (x>y) {t=x;
              x=y;
              y=t;
            }
    if (y>z) {t=z;
              z=y;
            }
    if (x<t) y=t;
    else
        { y=x;
          x=t;
        }
    printf("%d%d%d\n", x, y, z);
}/*sort*/
```

从以上算法可知需要进行 3 次比较, 在最坏情况下需移动记录 7 次。

6. 猴子吃桃问题。

猴子第一天摘下若干个桃子, 当即吃了一半, 还不过瘾, 又多吃了一个; 第二天早上又将剩下的桃子吃掉一半, 又多吃了一个; 以后每天早上都吃了前一天剩下的一半零一个, 到第 10 天早上想再吃时, 见只剩下一个桃子了。求第一天共摘多少个桃子。

解答:

```
main( )
{ int day, x1, x2;
  day=9;
  x2=1;
  while(day>0)
  { x1=(x2+1)*2;
    x2=x1;
    day - -;
  }
  printf("桃子总数=%d\n", x1);
}/*main*/
```

结果: 桃子总数为 1534。

7. 分析下列程序段, 求用大 O 记号表示执行时间为 n 的函数。

① $i=1; k=0;$

```
while(i<=n-1)
{ k=k*10*i;
  i++;
}
```

解答: $T(n)=O(n)$

```
② i=1; j=0;
   while( (i+j) <=n)
       if (i>j) j++;
       else i++;
```

解答: $T(n)=O(n)$

```
③ m=91; n=100;
   while(n>0)
       if(m>0)
           {m=m-10;
            n=n-1;
           }
       else
           m=m+1;
```

解答: $T(n)=O(n)$

```
④ for(i=0; i<n; i++)
    for(j=0; j<i; j++)
        for(k=0; k<j; k++)
            x=x+1;
```

解答: $T(n)=O(n^3)$

```
⑤ i=1;
   do
       { j=1;
         do
             { printf("%d\n", i*j);
               j++;
             }while(j>n);
         i++;
       }while(i>n);
```

解答: $T(n)=n(n+1)/2=O(n^2)$

```
⑥ x=n; /*n>1*/
   y=0;
   while((x>=(y+1)*(y+1))
       y=y+1;
```

解答: $T(n)=O(\sqrt{n})$

第二章 线性表

2.1 基础知识

一、填空题

1. 顺序存储的线性表，设其长度为 n ，在任何位置上插入或删除操作的时间代价基本上都是等效的。则插入一个元素大约要移动表中的 ① 个元素，删除一个元素时大约要移动表中的 ② 个元素。

2. 数组的长度是 ①，线性表的长度是 ②。

3. 当向一个顺序表插入一个元素时，从插入位置开始向后的所有元素均需 ① 一个位置，移动过程是从 ② 向 ③ 依次移动每一个元素。

4. 要从一个顺序表删除一个元素时，被删除元素之后的所有元素均需 ① 一个位置，移动过程是从 ② 向 ③ 依次移动每一个元素。

5. 向顺序表中第 i 个元素之前插入一个新元素时，首先从 ① 开始向后的所有元素均需 ② 一个位置，接着把新元素写入 ③ 上，最后使线性表的长度 ④，从顺序表中删除第 i 个元素时，首先把第 i 个元素赋给 ⑤，接着从 ⑥ 开始向后，所有元素均 ⑦ 一个位置，最后使线性表的长度 ⑧。

6. 在线性表的顺序存储中，元素之间的逻辑关系是通过 ① 决定的；在线性表的链接存储中，元素之间的逻辑关系是通过 ② 决定的。

7. 对于一个具有 n 个结点的单链表，在已知的结点 $*p$ 后插入一个新结点的时间复杂度为 ①，在给定值为 x 的结点后插入一个新结点的时间复杂度为 ②。

8. 在双向链表中，每个结点含有两个指针域，一个指向 ① 结点，另一个指向 ② 结点。

9. 在一个单链表中删除 $*p$ 结点时，应执行下列操作：

```
q=p->next;
p->data=p->next->data;
p->next= ① ;
free(q);
```

10. 若要在一个不带表头结点的单链表的首结点 $*p$ 结点之前插入一个 $*s$ 结点时，可执行下列操作：

```
s->next= ① ;
p->next=s;
t=p->data;
p->data= ② ;
s->data= ③ ;
```

11. 根据线性表的链式存储结构中每个结点所含指针的个数, 链表可分为 ① 和 ②; 而根据指针的联接方式, 链表又可分为 ③ 和 ④。

12. 对于线性表的顺序存储, 需要预先分配好存储空间, 若分配太多容易造成存储空间的 ①, 若分配太少又容易在算法中造成 ②, 因而只适用于数据量变化不大的情况; 对于线性表的链接存储, 不需要 ③ 存储空间, 存储器中的整个 ④ 都可供使用, 分配和回收结点都非常方便, 能够有效地利用存储空间, 在算法中不必考虑 ⑤ 的发生, 因而适用于数据量变化较大的情况。

13. 当对一个线性表经常进行的是存取操作, 而很少进行插入和删除操作时, 则采用 ① 存储结构为宜, 相反, 当经常进行的是插入和删除操作时, 则采用 ② 存储结构为宜。

14. 在单链表中设置头结点的作用是 ①。

15. 顺序表中逻辑上相邻的元素, 物理位置 ① 紧邻, 单链表中逻辑上相邻的元素, 物理位置 ② 紧邻。

二、选择题

1. 在一个长度为 n 的顺序表中删除第 i 个元素 ($0 < i < n$) 时, 需向前移动___个元素。

- (A) $n-i$ (B) $n-i+1$ (C) $n-i-1$ (D) i

2. 从一个具有 n 个结点的单链表中查找其值等于 x 的结点时, 在查找成功的情况下, 需平均比较___个元素结点。

- (A) $n/2$ (B) n (C) $(n+1)/2$ (D) $(n-1)/2$

3. 对一个具有 n 个元素的线性表, 建立其单链表的时间复杂度为_____。

- (A) $O(n)$ (B) $O(1)$ (C) $O(n^2)$ (D) $O(\log_2 n)$

4. 在双向循环链表中, 在 p 所指的结点之后插入 s 指针所指的结点, 其操作是_____。

- (A) $p \rightarrow next = s;$ $s \rightarrow prior = p;$
 $(p \rightarrow next) \rightarrow prior = s;$ $s \rightarrow next = p \rightarrow next;$
 (B) $s \rightarrow prior = p;$ $s \rightarrow next = p \rightarrow next;$
 $p \rightarrow next = s;$ $p \rightarrow next \rightarrow prior = s;$
 (C) $p \rightarrow next = s;$ $p \rightarrow next \rightarrow prior = s;$
 $s \rightarrow prior = p;$ $s \rightarrow next = p \rightarrow next;$
 (D) $s \rightarrow prior = p;$ $s \rightarrow next = p \rightarrow next;$
 $p \rightarrow next \rightarrow prior = s;$ $p \rightarrow next = s;$

5. 设单链表中指针 p 指向结点 m , 若要删除 m 之后的结点 (若存在), 则需修改指针的操作为_____。

- (A) $p \rightarrow next = p \rightarrow next \rightarrow next;$ (B) $p = p \rightarrow next;$
 (C) $p = p \rightarrow next \rightarrow next;$ (D) $p \rightarrow next = p;$

6. 线性表采用链式存储时, 其地址_____。

- (A) 必须是连续的
 (B) 一定是不连续的
 (C) 部分地址必须是连续的
 (D) 连续与否均可以

7. 在一个具有 n 个结点的有序单链表中插入一个新结点使得仍然有序, 其算法的时间复杂度为_____。

- (A) $O(\log_2 n)$ (B) $O(1)$ (C) $O(n^2)$ (D) $O(n)$

8. 在一个单链表中, 已知 $*q$ 结点是 $*p$ 结点的前趋结点, 若在 $*q$ 和 $*p$ 之间插入 $*s$ 结点, 则须执行_____。

- (A) $s \rightarrow \text{next} = p \rightarrow \text{next}; p \rightarrow \text{next} = s$
 (B) $q \rightarrow \text{next} = s; s \rightarrow \text{next} = p$
 (C) $p \rightarrow \text{next} = s \rightarrow \text{next}; s \rightarrow \text{next} = p$
 (D) $p \rightarrow \text{next} = s; s \rightarrow \text{next} = q$

9. 线性表是_____。

- (A) 一个有限序列, 可以为空
 (B) 一个有限序列, 不可以为空
 (C) 一个无限序列, 可以为空
 (D) 一个无限序列, 不可以为空

10. 在一个长度为 n 的顺序表中向第 i 个元素 ($0 < i \leq n+1$) 之前插入一个新元素时, 需向后移动_____个元素。

- (A) $n-i$ (B) $n-i+1$ (C) $n-i-1$ (D) i

解答:

一、填空题

- ① $(n+1)/2$ ② $(n-1)/2$
- ① 数组元素的个数 ② 表中数据元素的个数
- ① 后移 ② 后 ③ 前
- ① 前移 ② 前 ③ 后
- ① 位置 i ② 后移 ③ 位置 i ④ 加 1 ⑤ 工作单元 ⑥ 位置 $i+1$ ⑦ 前移
- ⑧ 减 1
- ① 物理存储位置 ② 链域的指针值
- ① $O(1)$ ② $O(n)$
- ① 前趋 ② 后继
- ① $q \rightarrow \text{next}$
- ① $p \rightarrow \text{next}$ ② $s \rightarrow \text{data}$ ③ t
- ① 单链表 ② 双链表 ③ 非循环链表 ④ 循环链表
- ① 浪费 ② 上溢 ③ 预先分配 ④ 空间 ⑤ 上溢
- ① 顺序 ② 链接
- ① 使空表和非空表统一; 算法处理一致。
- ① 一定 ② 不一定

二、选择题

1. A 2. C 3. A 4. D 5. A

6. D 7. D 8. B 9. A 10. B

2.2 应用知识

1. 描述以下三个概念的区别：头指针，头结点，表头结点。

解答：头指针是指向链表中第一个结点（即表头结点）的指针，在表头结点之前附设的一个结点称为头结点；表头结点为链表中存储线性表中第一个数据元素的结点。若链表中附设头结点，则不管线性表是否为空表，头指针均不为空，否则表示空表的链表的头指针为空。

2. 在单链表和双向链表中，能否从当前结点出发访问任一结点？

解答：在单链表中只能由当前结点访问其后的任一结点，因为没有指向其前趋结点的指针。而在双向链表中既有指向其后继结点的指针又有指向其前趋结点的指针，故可由当前结点出发访问链表中的任一结点。

3. 线性表的两种存储结构各有哪些优缺点？

解答：线性表具有两种存储结构即顺序存储结构和链接存储结构。线性表的顺序存储结构可以直接存取数据元素，方便灵活、效率高，但插入、删除操作时将会引起元素的大量移动，因而降低效率；而在链接存储结构中内存采用动态分配，利用率高，但需增设指示结点之间关系的指针域，存取数据元素不如顺序存储方便，但结点的插入、删除操作较简单。

4. 对于线性表的两种存储结构，如果有 n 个线性表同时并存，而且在处理过程中各表的长度会动态发生变化，线性表的总数也会自动改变，在此情况下，应选用哪一种存储结构？为什么？

解答：应选用链接存储结构，因为链式存储结构是用一组任意的存储单元依次存储线性表中的各元素，这里存储单元可以是连续的，也可以是不连续的；这种存储结构对于元素的删除或插入运算是需要移动元素的，只需修改指针即可，所以很容易实现表的容量的扩充。

5. 对于线性表的两种存储结构，若线性表的总数基本稳定，且很少进行插入和删除操作，但要求以最快的速度存取线性表中的元素，那么应选用何种存储结构？试说明理由。

解答：应选用顺序存储结构，因为每个数据元素的存储位置和线性表的起始位置相差一个和数据元素在线性表中的序号成正比的常数。因此，只要确定了其起始位置，线性表中的任一个数据元素都可随机存取，因此，线性表的顺序存储结构是一种随机存取的存储结构，而链表则是一种顺序存取的存储结构。

6. 多项式 $T(x)=a_nx^n+a_{n-1}x^{n-1}+\cdots+a_1x+a_0$ 的线性表表示法有下列两种可能的形式：

$$(1) A=(n, a_n, a_{n-1}, \cdots, a_1, a_0)$$

$$(2) a=(m, e_{m-1}, c_{m-1}, e_{m-2}, c_{m-2}, \cdots, e_1, c_1, e_0, c_0)$$

其中， m 为非零项的个数， e_i, c_i 分别为非零项的指数和系数，试问：要进行多项式相加采用哪一种方法处理较为简单。

解答：由已知条件可知，用第 (1) 种表示方法需要 $n+2$ 个实数存储单元，这里 n 是多项式的最高幂次；而用第 (2) 种表示方法需用 $2m+1$ 个实数存储单元，其中 m 为非零系数的个数。显然，当非零系数 m 较少时 ($m < (n+1)/2$)，用第 (2) 种表示方法只需要较少的存储空间；但若进行多项式相加，此时用第 1 种表示法处理较为简单，只需将次数较低的多项式的各项系数加到次数较高的多项式的相应项的系数上去即可；而用第 (2) 种方法虽可减