关注微信公众号【神灯考研】，获取更多考研资源！

目　录

微信公众号【神灯考研】
考研人的精神家园
微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451
考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

微信公众号【神灯考研】
考研人的精神家园
微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451
考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

微信公众号【神灯考研】
考研人的精神家园
微信公众号：神灯考研 客服微信：KYFT104 QQ群：118105451
考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

微信公众号【神灯考研】
考研人的精神家园
微信公众号：神灯考研 客服微信：KYFT104 QQ群：118105451
考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

微信公众号【神灯考研】
考研人的精神家园
微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451
考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

第1章 绪 论

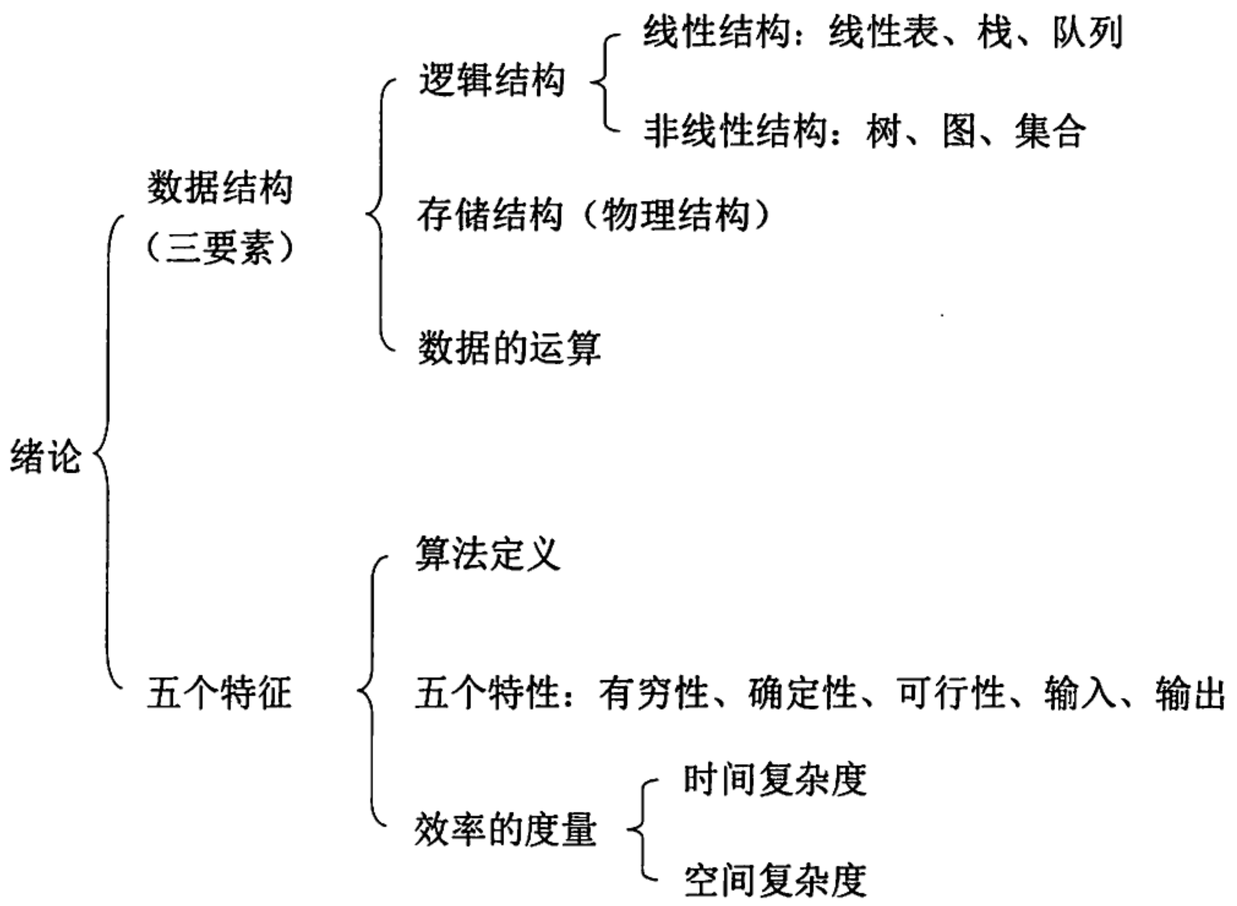
【考纲内容】

算法时间复杂度和空间复杂度的分析与计算

兑换后

看视频讲解

【知识框架】



【复习提示】

本章内容是数据结构概述，并不在考研大纲中。读者可通过对本章的学习，初步了解数据结构的基本内容和基本方法。分析算法的时间复杂度和空间复杂度是本章的重点，一定要熟练掌握，算法设计题通常都会要求分析时间复杂度、空间复杂度，同时会出现考查时间复杂度的选择题。

1.1 数据结构的基本概念

1.1.1 基本概念和术语

1．数据

数据是信息的载体，是描述客观事物属性的数、字符及所有能输入到计算机中并被计算机程序识别和处理的符号的集合。数据是计算机程序加工的原料。

2．数据元素

数据元素是数据的基本单位，通常作为一个整体进行考虑和处理。一个数据元素可由若干数据项组成，数据项是构成数据元素的不可分割的最小单位。例如，学生记录就是一个数据元素，它由学号、姓名、性别等数据项组成。

微信公众号【神灯考研】
考研人的精神家园
微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451
考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

3．数据对象

数据对象是具有相同性质的数据元素的集合，是数据的一个子集。例如，整数数据对象是集合 $N=\{0,\pm1,\pm2,\cdots\}$。

4．数据类型

数据类型是一个值的集合和定义在此集合上的一组操作的总称。

1）原子类型。其值不可再分的数据类型。

2）结构类型。其值可以再分解为若干成分（分量）的数据类型。

3）抽象数据类型。抽象数据组织及与之相关的操作。

5．数据结构

数据结构是相互之间存在一种或多种特定关系的数据元素的集合。在任何问题中，数据元素都不是孤立存在的，它们之间存在某种关系，这种数据元素相互之间的关系称为结构（Structure）。数据结构包括三方面的内容：逻辑结构、存储结构和数据的运算。

数据的逻辑结构和存储结构是密不可分的两个方面，一个算法的设计取决于所选定的逻辑结构，而算法的实现依赖于所采用的存储结构[①]。

1.1.2　数据结构三要素

1．数据的逻辑结构

逻辑结构是指数据元素之间的逻辑关系，即从逻辑关系上描述数据。它与数据的存储无关，是独立于计算机的。数据的逻辑结构分为线性结构和非线性结构，线性表是典型的线性结构；集合、树和图是典型的非线性结构。数据的逻辑结构分类如图 1.1 所示。

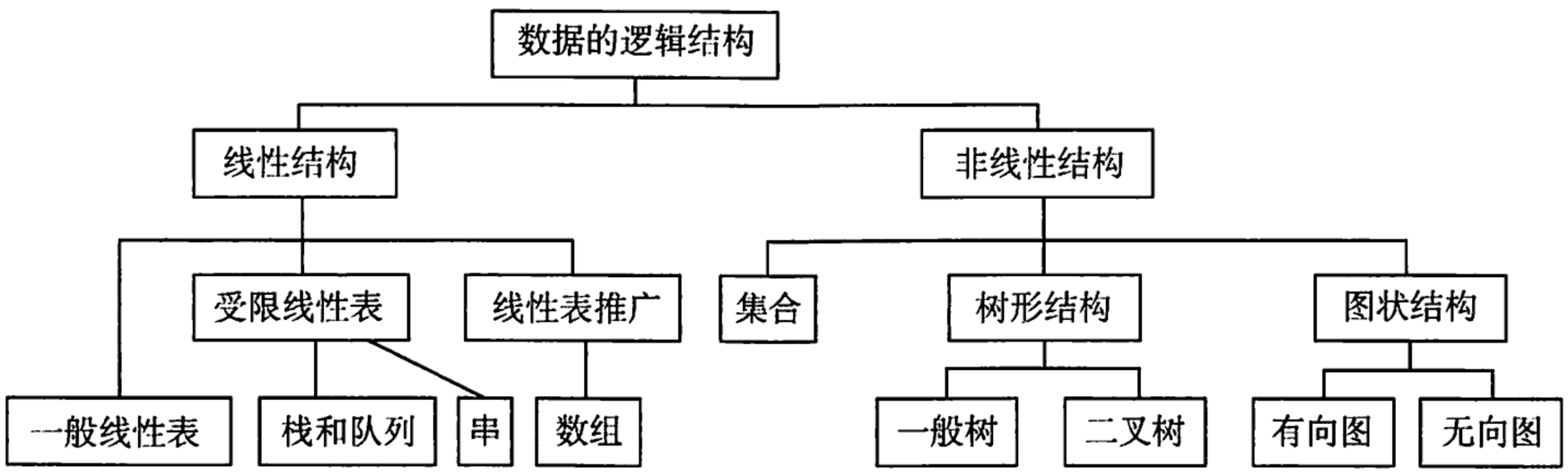


图 1.1　数据的逻辑结构分类图

集合。结构中的数据元素之间除“同属一个集合”外，别无其他关系，如图 1.2(a)所示。

线性结构。结构中的数据元素之间只存在一对一的关系，如图 1.2(b)所示。

树形结构。结构中的数据元素之间存在一对多的关系，如图 1.2(c)所示。

图状结构或网状结构。结构中的数据元素之间存在多对多的关系，如图 1.2(d)所示。

2．数据的存储结构

存储结构是指数据结构在计算机中的表示（又称映像），也称物理结构。它包括数据元素的表示和关系的表示。数据的存储结构是用计算机语言实现的逻辑结构，它依赖于计算机语言。数据的存储结构主要有顺序存储、链式存储、索引存储和散列存储。

微信公众号【神灯考研】
考研人的精神家园

① 读者应通过后续章节的学习，逐步理解设计与实现的概念与区别。

微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451

考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

1）*顺序存储*。把逻辑上相邻的元素存储在物理位置上也相邻的存储单元中，元素之间的关系由存储单元的邻接关系来体现。其优点是可以实现随机存取，每个元素占用最少的存储空间；缺点是只能使用相邻的一整块存储单元，因此可能产生较多的外部碎片。

2）*链式存储*。不要求逻辑上相邻的元素在物理位置上也相邻，借助指示元素存储地址的指针来表示元素之间的逻辑关系。其优点是不会出现碎片现象，能充分利用所有存储单元；缺点是每个元素因存储指针而占用额外的存储空间，且只能实现顺序存取。

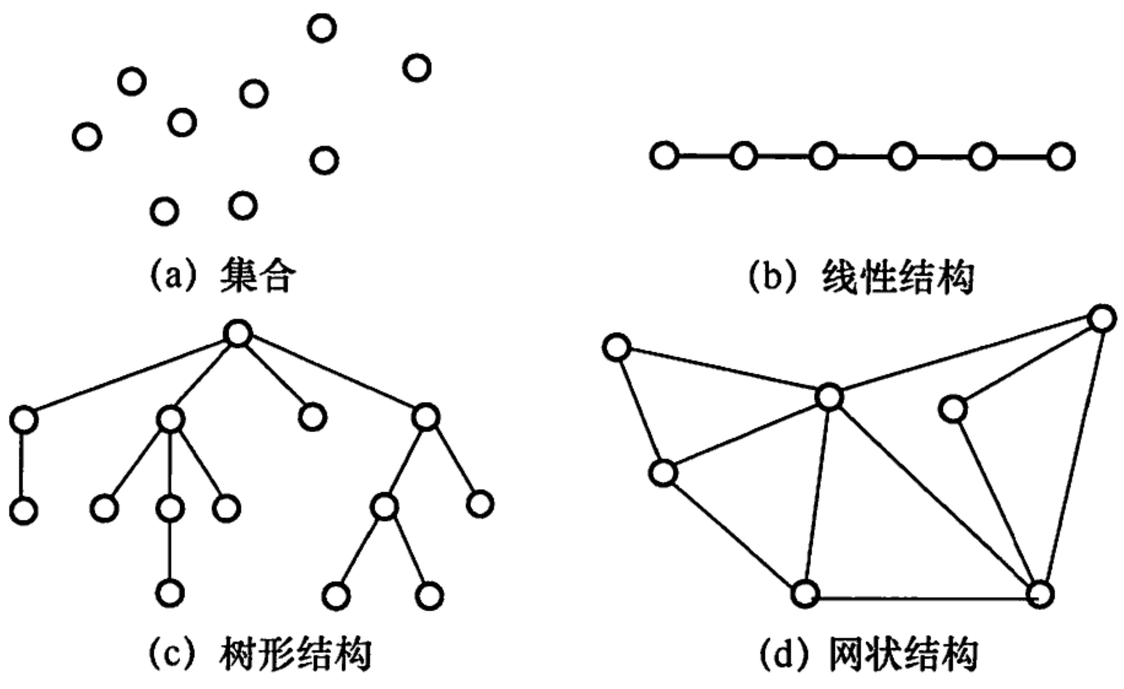


图 1.2　4 类基本结构关系示例图

3）*索引存储*。在存储元素信息的同时，还建立附加的索引表。索引表中的每项称为*索引项*，索引项的一般形式是（关键字，地址）。其优点是检索速度快；缺点是附加的索引表额外占用存储空间。另外，增加和删除数据时也要修改索引表，因而会花费较多的时间。

4）*散列存储*。根据元素的关键字直接计算出该元素的存储地址，又称哈希（Hash）*存储*。其优点是检索、增加和删除结点的操作都很快；缺点是若散列函数不好，则可能出现元素存储单元的冲突，而解决冲突会增加时间和空间开销。

3．数据的运算

施加在数据上的运算包括运算的定义和实现。运算的定义是针对逻辑结构的，指出运算的功能；运算的实现是针对存储结构的，指出运算的具体操作步骤。

1.1.3　本节试题精选

一、单项选择题

01. 可以用（　）定义一个完整的数据结构。

A. 数据元素　　B. 数据对象　　C. 数据关系　　D. 抽象数据类型

02. 以下数据结构中，（　）是非线性数据结构。

A. 树　　B. 字符串　　C. 队列　　D. 栈

03. 以下属于逻辑结构的是（　）。

A. 顺序表　　B. 哈希表　　C. 有序表　　D. 单链表

04. 以下与数据的存储结构无关的术语是（　）。

A. 循环队列　　B. 链表　　C. 哈希表　　D. 栈

05. 以下关于数据结构的说法中，正确的是（　）。

A. 数据的逻辑结构独立于其存储结构

B. 数据的存储结构独立于其逻辑结构

C. 数据的逻辑结构唯一决定其存储结构

D. 数据结构仅由其逻辑结构和存储结构决定

微信公众号【神灯考研】
考研人的精神家园

06. 在存储数据时，通常不仅要存储各数据元素的值，而且要存储（　）。

A. 数据的操作方法　　B. 数据元素的类型

C. 数据元素之间的关系　　D. 数据的存取方法

微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451

考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

07. 链式存储设计时，结点内的存储单元地址（　）。

A. 一定连续　　B. 一定不连续

C. 不一定连续　　D. 部分连续，部分不连续

二、综合应用题

01. 对于两种不同的数据结构，逻辑结构或物理结构一定不相同吗？

02. 试举一例，说明对相同的逻辑结构，同一种运算在不同的存储方式下实现时，其运算效率不同。

1.1.4　答案与解析

一、单项选择题

01. D

抽象数据类型（ADT）描述了数据的逻辑结构和抽象运算，通常用（数据对象，数据关系，基本操作集）这样的三元组来表示，从而构成一个完整的数据结构定义。

02. A

树和图是典型的非线性数据结构，其他选项都属于线性数据结构。

03. C

顺序表、哈希表和单链表是三种不同的数据结构，既描述逻辑结构，又描述存储结构和数据运算。而有序表是指关键字有序的线性表，仅描述元素之间的逻辑关系，它既可以链式存储，又可以顺序存储，故属于逻辑结构。

04. D

数据的存储结构有顺序存储、链式存储、索引存储和散列存储。循环队列（易错点）是用顺序表表示的队列，是一种数据结构。栈是一种抽象数据类型，可采用顺序存储或链式存储，只表示逻辑结构。

05. A

数据的逻辑结构是从面向实际问题的角度出发的，只采用抽象表达方式，独立于存储结构，数据的存储方式有多种不同的选择；而数据的存储结构是逻辑结构在计算机上的映射，它不能独立于逻辑结构而存在。数据结构包括三个要素，缺一不可。

06. C

在存储数据时，不仅要存储数据元素的值，而且要存储数据元素之间的关系。

07. A

链式存储设计时，各个不同结点的存储空间可以不连续，但结点内的存储单元地址必须连续。

二、综合应用题

01.【解答】

应该注意到，数据的运算也是数据结构的一个重要方面。

对于两种不同的数据结构，它们的逻辑结构和物理结构完全有可能相同。比如二叉树和二叉排序树，二叉排序树可以采用二叉树的逻辑表示和存储方式，前者通常用于表示层次关系，而后者通常用于排序和查找。虽然它们的运算都有建立树、插入结点、删除结点和查找结点等功能，但对于二叉树和二叉排序树，这些运算的定义是不同的，以查找结点为例，二叉树的时间复杂度为 $O(n)$，而二叉排序树的时间复杂度为 $O(\log_2 n)$。

02.【解答】

微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451

考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

线性表既可以用顺序存储方式实现，又可以用链式存储方式实现。在顺序存储方式下，在线性表中插入和删除元素，平均要移动近一半的元素，时间复杂度为 $O(n)$；而在链式存储方式下，插入和删除的时间复杂度都是 $O(1)$。

1.2　算法和算法评价

1.2.1　算法的基本概念

算法（Algorithm）是对特定问题求解步骤的一种描述，它是指令的有限序列，其中的每条指令表示一个或多个操作。此外，一个算法还具有下列 5 个重要特性：

1）有穷性。一个算法必须总在执行有穷步之后结束，且每一步都可在有穷时间内完成。

2）确定性。算法中每条指令必须有确切的含义，对于相同的输入只能得出相同的输出。

3）可行性。算法中描述的操作都可以通过已经实现的基本运算执行有限次来实现。

4）输入。一个算法有零个或多个输入，这些输入取自于某个特定的对象的集合。

5）输出。一个算法有一个或多个输出，这些输出是与输入有着某种特定关系的量。

通常，设计一个“好”的算法应考虑达到以下目标：

1）正确性。算法应能够正确地解决求解问题。

2）可读性。算法应具有良好的可读性，以帮助人们理解。

3）健壮性。输入非法数据时，算法能适当地做出反应或进行处理，而不会产生莫名其妙的输出结果。

4）高效率与低存储量需求。效率是指算法执行的时间，存储量需求是指算法执行过程中所需要的最大存储空间，这两者都与问题的规模有关。

1.2.2　算法效率的度量

算法效率的度量是通过时间复杂度和空间复杂度来描述的。

1．时间复杂度

一个语句的频度是指该语句在算法中被重复执行的次数。算法中所有语句的频度之和记为 $T(n)$，它是该算法问题规模 n 的函数，时间复杂度主要分析 $T(n)$的数量级。算法中基本运算（最深层循环内的语句）的频度与 $T(n)$同数量级，因此通常采用算法中基本运算的频度 $f(n)$来分析算法的时间复杂度[①]。因此，算法的时间复杂度记为

$$T(n) = O(f(n))$$

式中，O 的含义是 $T(n)$的数量级，其严格的数学定义是：若 $T(n)$和 $f(n)$是定义在正整数集合上的两个函数，则存在正常数 C 和 n_0，使得当 $n \geqslant n_0$ 时，都满足 $0 \leqslant T(n) \leqslant Cf(n)$。

算法的时间复杂度不仅依赖于问题的规模 n，也取决于待输入数据的性质（如输入数据元素的初始状态）。例如，在数组 A[0…n-1]中，查找给定值 k 的算法大致如下：

```
(1) i=n-1;
(2) while(i>=0&&(A[i]!=k))
(3)     i--;
(4) return i;
```

微信公众号【神灯考研】
考研人的精神家园

① 取 $f(n)$中随 n 增长最快的项，将其系数置为 1 作为时间复杂度的度量。例如，$f(n) = an^3 + bn^2 + cn$ 的时间复杂度为 $O(n^3)$。

微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451

考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

该算法中语句 3（基本运算）的频度不仅与问题规模 n 有关，而且与输入实例中 A 的各元素的取值及 k 的取值有关：

① 若 A 中没有与 k 相等的元素，则语句 3 的频度 $f(n)=n$。

② 若 A 的最后一个元素等于 k，则语句 3 的频度 $f(n)$是常数 0。

最坏时间复杂度是指在最坏情况下，算法的时间复杂度。

平均时间复杂度是指所有可能输入实例在等概率出现的情况下，算法的期望运行时间。

最好时间复杂度是指在最好情况下，算法的时间复杂度。

一般总是考虑在最坏情况下的时间复杂度，以保证算法的运行时间不会比它更长。

在分析一个程序的时间复杂性时，有以下两条规则：

a）加法规则

$$T(n)=T_1(n)+T_2(n)=O(f(n))+O(g(n))=O(\max(f(n),g(n)))$$

b）乘法规则

$$T(n)=T_1(n)\times T_2(n)=O(f(n))\times O(g(n))=O(f(n)\times g(n))$$

常见的渐近时间复杂度为

$$O(1)<O(\log_2 n)<O(n)<O(n\log_2 n)<O(n^2)<O(n^3)<O(2^n)<O(n!)<O(n^n)$$

2．空间复杂度

算法的空间复杂度 $S(n)$定义为该算法所耗费的存储空间，它是问题规模 n 的函数。记为

$$S(n)=O(g(n))$$

一个程序在执行时除需要存储空间来存放本身所用的指令、常数、变量和输入数据外，还需要一些对数据进行操作的工作单元和存储一些为实现计算所需信息的辅助空间。若输入数据所占空间只取决于问题本身，和算法无关，则只需分析除输入和程序之外的额外空间。

算法原地工作是指算法所需的辅助空间为常量，即 $O(1)$。

1.2.3　本节试题精选

一、单项选择题

01. 一个算法应该是（　）。

A. 程序　　B. 问题求解步骤的描述

C. 要满足五个基本特性　　D. A 和 C

02. 某算法的时间复杂度为 $O(n^2)$，表明该算法的（　）。

A. 问题规模是 n^2　　B. 执行时间等于 n^2

C. 执行时间与 n^2 成正比　　D. 问题规模与 n^2 成正比

03. 以下算法的时间复杂度为（　）。

```
void fun(int n){
    int i=1;
    while(i<=n)
        i=i*2;
}
```

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

04. 有以下算法，其时间复杂度为（　）。

```
void fun(int n){
    int i=0;
    while(i*i*i<=n)
```

微信公众号【神灯考研】
考研人的精神家园
微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451
考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！


```
            i++;
        }
```

A. $O(n)$　　B. $O(n\log n)$　　C. $O(\sqrt[3]{n})$　　D. $O(\sqrt{n})$

05. 程序段如下：

```
for(i=n-1;i>1;i--)
    for(j=1;j<i;j++)
        if(A[j]>A[j+1])
            A[j]与A[j+1]对换;
```

其中 n 为正整数，则最后一行语句的频度在最坏情况下是（　）。

A. $O(n)$　　B. $O(n\log n)$　　C. $O(n^3)$　　D. $O(n^2)$

06. 以下算法中加下画线的语句的执行次数为（　）。

```
int m=0,i,j;
for(i=1;i<=n;i++)
    for(j=1;j<=2*i;j++)
        m++;
```

A. $n(n+1)$　　B. n　　C. $n+1$　　D. n^2

07. 【2011 统考真题】设 n 是描述问题规模的非负整数，下面的程序片段的时间复杂度是（　）。

```
x=2;
while(x<n/2)
x=2*x;
```

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

08. 【2012 统考真题】求整数 n（$n\geqslant 0$）的阶乘的算法如下，其时间复杂度是（　）。

```
int fact(int n){
    if(n<=1) return 1;
    return n*fact(n-1);
}
```

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

09. 【2013 统考真题】已知两个长度分别为 m 和 n 的升序链表，若将它们合并为长度为 $m+n$ 的一个降序链表，则最坏情况下的时间复杂度是（　）。

A. $O(n)$　　B. $O(mn)$　　C. $O(\min(m,n))$　　D. $O(\max(m,n))$

10. 【2014 统考真题】下列程序段的时间复杂度是（　）。

```
count=0;
for(k=1;k<=n;k*=2)
    for(j=1;j<=n;j++)
        count++;
```

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

11. 【2017 统考真题】下列函数的时间复杂度是（　）。

```
int func(int n){
    int i=0, sum=0;
    while(sum<n) sum += ++i;
    return i;
}
```

A. $O(\log n)$　　B. $O(n^{1/2})$　　C. $O(n)$　　D. $O(n\log n)$

12. 【2019 统考真题】设 n 是描述问题规模的非负整数，下列程序段的时间复杂度是（　）。

微信公众号【神灯考研】
考研人的精神家园
微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451
考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

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

A. $O(\log n)$　　B. $O(n^{1/2})$　　C. $O(n)$　　D. $O(n^2)$

13. 【2022统考真题】下列程序段的时间复杂度是（　）。

```
int sum=0;
for(int i=1;i<n;i*=2)
     for(int j=0;j<i;j++)
          sum++;
```

A. $O(\log n)$　　B. $O(n)$　　C. $O(n\log n)$　　D. $O(n^2)$

二、综合应用题

01. 一个算法所需时间由下述递归方程表示，试求出该算法的时间复杂度的级别（或阶）。

$$T(n)=\begin{cases}1, & n=1\\ 2T(n/2)+n, & n>1\end{cases}$$

式中，n是问题的规模，为简单起见，设n是2的整数次幂。

02. 分析以下各程序段，求出算法的时间复杂度。

①
```
i=1;k=0;
while(i<n-1){
     k=k+10*i;
     i++;
}
```

②
```
y=0;
while((y+1)*(y+1)<=n)
     y=y+1;
```

③
```
for(i=1;i<=n;i++)
     for(j=1;j<=i;j++)
          for(k=1;k<=j;k++)
               x++;
```

④
```
for(i=0;i<n;i++)
     for(j=0;j<m;j++)
          a[i][j]=0;
```

1.2.4　答案与解析

一、单项选择题

01. B

本题是中山大学考研真题，题目本身没有问题，考查的是算法的定义。程序不一定满足有穷性，如死循环、操作系统等，而算法必须有穷。算法代表对问题求解步骤的描述，而程序则是算法在计算机上的特定实现。不少读者认为C也对，它只是算法的必要条件，不能成为算法的定义。

02. C

时间复杂度为$O(n^2)$，说明算法的时间复杂度$T(n)$满足$T(n)\leqslant cn^2$（c为比例常数），即$T(n)=O(n^2)$，时间复杂度$T(n)$是问题规模n的函数，其问题规模仍然是n而不是n^2。

03. D

找出基本运算`i=i*2`，设执行次数为t，则$2^t\leqslant n$，即$t\leqslant\log_2 n$，因此时间复杂度$T(n)=O(\log_2 n)$。

更直观的方法：计算基本运算`i=i*2`的执行次数（每执行一次`i`乘2），其中判断条件可理解为$2^t=n$，即$t=\log_2 n$，则$T(n)=O(\log_2 n)$。（注意，本方法可灵活运用到第4题和第8题。）

04. C

基本运算为`i++`，设执行次数为t，有$t\times t\times t\leqslant n$，即$t^3\leqslant n$。故有$t\leqslant\sqrt[3]{n}$，则$T(n)=O(\sqrt[3]{n})$。

05. D

这是冒泡排序的算法代码，考查最坏情况下的元素交换次数（若觉得理解困难可在学完第8

微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451
考研人的精神家园
考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

章后再回顾）。当所有相邻元素都为逆序时，则最后一行的语句每次都会执行。此时，

$$T(n)=\sum_{i=2}^{n-1}\sum_{j=1}^{i-1}1=\sum_{i=2}^{n-1}i-1=(n-2)(n-1)/2=O(n^2)$$

所以在最坏情况下的该语句频度是 $O(n^2)$。

06．A

m++语句的执行次数为

$$\sum_{i=1}^{n}\sum_{j=1}^{2i}1=\sum_{i=1}^{n}2i=2\sum_{i=1}^{n}i=n(n+1)$$

07．A

基本运算（执行频率最高的语句）为 x=2*x，每执行一次 x 乘 2，设执行次数为 t，则有 $2^{t+1}<n/2$，所以 $t<\log_2(n/2)-1=\log_2 n-2$，得 $T(n)=O(\log_2 n)$。

08．B

本题是求阶乘 $n!$的递归代码，即 $n\times(n-1)\times\cdots\times1$。每次递归调用时 fact()的参数减 1，递归出口为 fact(1)，一共执行 n 次递归调用 fact()，故 $T(n)=O(n)$。

09．D

两个升序链表合并，两两比较表中元素，每比较一次，确定一个元素的链接位置（取较小元素，头插法）。当一个链表比较结束后，将另一个链表的剩余元素插入即可。最坏的情况是两个链表中的元素依次进行比较，因为 $2\max(m,n)\geqslant m+n$，所以时间复杂度为 $O(\max(m,n))$。

10．C

内层循环条件 $j\leqslant n$ 与外层循环的变量无关，各自独立，每执行一次 j 自增 1，每次内层循环都执行 n 次。外层循环条件 $k\leqslant n$，增量定义为 k*=2，可知循环次数 t 满足 $k=2^t\leqslant n$，即 $t\leqslant\log_2 n$。即内层循环的时间复杂度为 $O(n)$，外层循环的时间复杂度为 $O(\log_2 n)$。对于嵌套循环，根据乘法规则可知，该段程序的时间复杂度 $T(n)=T_1(n)\times T_2(n)=O(n)\times O(\log_2 n)=O(n\log_2 n)$。

11．B

基本运算 sum+=++i，它等价于++i;sum=sum+i，每执行一次 i 自增 1。i=1 时 sum=0+1；i=2 时 sum=0+1+2；i=3 时 sum=0+1+2+3，以此类推得出 sum=0+1+2+3+…+i=(1+i)*i/2，可知循环次数 t 满足 (1+t)*t/2<n，因此时间复杂度为 $O(n^{1/2})$。

注意：统考真题中经常把 $\log_2$ 书写为 log，此时默认底数为 2。

12．B

假设第 k 次循环终止，则第 k 次执行时，$(x+1)^2>n$，x 的初始值为 0，第 k 次判断时，$x=k-1$，即 $k^2>n$，$k>\sqrt{n}$，因此该程序段的时间复杂度为 $O(\sqrt{n})$。因此选 B。

13．B

当外层循环的变量 i 取不同值时，内层循环就执行多少次，因此总循环次数为 i 的所有取值之和。假设外层循环共执行 k 次，当 $i=1,2,4,8,\cdots,2^{k-1}(2^{k-1}<n\leqslant 2^k)$ 时，内层循环执行 i 次，因此总循环次数 $T=1+2+4+8+\cdots+2^{k-1}=2^k-1$，即 $n<T<2n$，时间复杂度为 $O(n)$。

微信公众号【神灯考研】
考研人的精神家园

二、综合应用题

01.【解答】

时间复杂度为 $O(n\log_2 n)$。

设 $n=2^k$（$k\geqslant 0$），根据题目所给定义有 $T(2^k)=2T(2^{k-1})+2^k=2^2T(2^{k-2})+2\times2^k$，由此可得一般递推公式 $T(2^k)=2^iT(2^{k-i})+i\times2^k$，进而得 $T(2^k)=2^kT(2^0)+k\times2^k=(k+1)2^k$，即 $T(n)=2^{\log_2 n}+\log_2 n\times n=$

微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451

考研冲刺密押课免费更新，请关注公众号：【最新考研资料】

关注微信公众号【神灯考研】，获取更多考研资源！

$n(\log_2 n+1)$，也就是 $O(n\log_2 n)$。

02.【解答】

① 基本语句 `k=k+10*i` 共执行了 $n-2$ 次，所以 $T(n)=O(n)$。

② 设循环体共执行 t 次，每循环一次，循环变量 y 加 1，最终 $t=y$。故 $t^2 \leqslant n$，得 $T(n)=O(n^{1/2})$。

③ 基本语句 `x++`的执行次数为 $T(n)=O\left(\sum_{i=1}^{n}\sum_{j=1}^{i}\sum_{k=1}^{j}1\right)=O\left(\frac{1}{6}n^3\right)=O(n^3)$。

④ 内循环执行 m 次，外循环执行 n 次，根据乘法原理，共执行了 $m\times n$ 次，故 $T(m,n)=O(m\times n)$。

归纳总结

本章的重点是分析程序的时间复杂度。一定要掌握分析时间复杂度的方法和步骤，很多读者在做题时一眼就能看出程序的时间复杂度，但就是无法规范地表述其推导过程。为此，编者查阅众多资料，总结出了此类题型的两种形式，供大家参考。

1．循环主体中的变量参与循环条件的判断

此类题应该找出主体语句中与 $T(n)$成正比的循环变量，将之代入条件中进行计算。例如，

```
1. int i=1;
   while(i<=n)
      i=i*2;.
```

```
2. int y=5;
   while((y+1)*(y+1)<n)
      y=y+1;.
```

例 1 中，i 乘以 2 的次数正是主体语句的执行次数 t，因此有 $2^t \leqslant n$，取对数后得 $t \leqslant \log_2 n$，则 $T(n)=O(\log_2 n)$。

例 2 中，y 加 1 的次数恰好与 $T(n)$成正比，记 t 为该程序的执行次数并令 $t=y-5$，有 $y=t+5$，$(t+5+1)\times(t+5+1)<n$，得 $t<\sqrt{n}-6$，即 $T(n)=O(\sqrt{n})$。

2．循环主体中的变量与循环条件无关

此类题可采用数学归纳法或直接累计循环次数。多层循环时从内到外分析，忽略单步语句、条件判断语句，只关注主体语句的执行次数。此类问题又可分为递归程序和非递归程序：

- 递归程序一般使用公式进行递推。例如习题 9 的时间复杂度分析如下：

$$T(n)=1+T(n-1)=1+1+T(n-2)=\cdots=n-1+T(1)$$

  即 $T(n)=O(n)$。
- 非递归程序比较简单，可以直接累计次数，例如习题 11 等。

思维拓展

求解斐波那契数列

$$F(n)=\begin{cases}0, & n=0\\ 1, & n=1\\ F(n-1)+F(n-2), & n>1\end{cases}$$

有两种常用的算法：递归算法和非递归算法。试分别分析两种算法的时间复杂度。（**提示：**请结合归纳总结中的两种方法进行解答。）

微信公众号【神灯考研】
考研人的精神家园
微信公众号：神灯考研　　客服微信：KYFT104　　QQ群：118105451
考研冲刺密押课免费更新，请关注公众号：【最新考研资料】