

高等学校教材·计算机科学与技术

可下载教学资料

<http://www.tup.tsinghua.edu.cn>

# 数据结构教程学习指导

李春葆 编著



清华大学出版社

高等学校教材·计算机科学与技术

# 数据结构教程学习指导

李春葆 编著

清华大学出版社  
北京

## 内 容 简 介

本书是《数据结构教程》的配套学习指导书,两书章次一一对应,内容包括线性表、栈和队列、串、数组和稀疏矩阵、递归、树状结构、广义表、图、查找、内排序、外排序、文件等。每一章中,除包含典型习题和教程中练习题及答案外,还包含补充练习题及参考答案。本书自成一体,可脱离《数据结构教程》单独使用。

本书适合作为高等院校计算机及其他相关系本科及研究生学习参考书。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

### 图书在版编目(CIP)数据

数据结构教程学习指导/李春葆编著. —北京:清华大学出版社,2005.7

(高等学校教材·计算机科学与技术)

ISBN 7-302-10863-3

I. 数… II. 李… III. 数据结构—高等学校—自学参考资料 IV. TP311.12

中国版本图书馆 CIP 数据核字(2005)第 038002 号

出版者:清华大学出版社 地 址:北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编:100084

社 总 机:010-62770175 客户服务:010-62776969

组稿编辑:丁 岭

文稿编辑:孙建春

印 装 者:北京鑫海金澳胶印有限公司

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:16.5 字数:387千字

版 次:2005年7月第1版 2005年7月第1次印刷

书 号:ISBN 7-302-10863-3/TP·7225

印 数:1~3000

定 价:22.00元

高等学校教材·计算机科学与技术

编审委员会成员

(按地区排序)

清华大学	周立柱	教授
	覃征	教授
	王建民	教授
	刘强	副教授
	冯建华	副教授
北京大学	杨冬青	教授
	陈钟	教授
	陈立军	副教授
北京航空航天大学	马殿富	教授
	吴超英	副教授
	姚淑珍	教授
中国人民大学	王珊	教授
	孟小峰	教授
	陈红	教授
北京交通大学	阮秋琦	教授
北京信息工程学院	孟庆昌	教授
北京科技大学	杨炳儒	教授
石油大学	陈明	教授
天津大学	艾德才	教授
复旦大学	吴立德	教授
	吴百锋	教授
	杨卫东	副教授
华东理工大学	邵志清	教授
华东师范大学	杨宗源	教授
	应吉康	教授
东华大学	乐嘉锦	教授
上海第二工业大学	蒋川群	教授
浙江大学	吴朝晖	教授
	李善平	教授
南京大学	骆斌	教授
南京航空航天大学	秦小麟	教授
南京理工大学	张功萱	教授

南京邮电学院	朱秀昌	教授
苏州大学	龚声蓉	教授
江苏大学	宋余庆	教授
武汉大学	何炎祥	教授
华中科技大学	刘乐善	教授
中南财经政法大学	刘腾红	教授
华中师范大学	王林平	副教授
	魏开平	教授
武汉理工大学	李中年	教授
国防科技大学	赵克佳	教授
	肖 依	副教授
中南大学	陈松乔	教授
湖南大学	林亚平	教授
	邹北骢	教授
西安交通大学	沈钧毅	教授
	齐 勇	教授
西北大学	周明全	教授
长安大学	巨永峰	教授
西安石油学院	方 明	教授
西安邮电学院	陈莉君	副教授
哈尔滨工业大学	郭茂祖	教授
吉林大学	徐一平	教授
	毕 强	教授
长春工程学院	沙胜贤	教授
山东大学	孟祥旭	教授
	郝兴伟	教授
山东科技大学	郑永果	教授
中山大学	潘小轰	教授
厦门大学	冯少荣	教授
福州大学	林世平	副教授
云南大学	刘惟一	教授
重庆邮电学院	王国胤	教授
西南交通大学	杨 燕	副教授

改革开放以来,特别是党的十五大以来,我国教育事业取得了举世瞩目的辉煌成就,高等教育实现了历史性的跨越,已由精英教育阶段进入国际公认的大众化教育阶段。在质量不断提高的基础上,高等教育规模取得如此快速的发展,创造了世界教育发展史上的奇迹。当前,教育工作既面临着千载难逢的良好机遇,同时也面临着前所未有的严峻挑战。社会不断增长的高等教育需求同教育供给特别是优质教育供给不足的矛盾,是现阶段教育发展面临的基本矛盾。

教育部一直十分重视高等教育质量工作。2001年8月,教育部下发了《关于加强高等学校本科教学工作,提高教学质量的若干意见》,提出了十二条加强本科教学工作提高教学质量的措施和意见。2003年6月和2004年2月,教育部分别下发了《关于启动高等学校教学质量与教学改革工程精品课程建设工作的通知》和《教育部实施精品课程建设提高高校教学质量和人才培养质量》文件,指出“高等学校教学质量和教学改革工程”是教育部正在制订的《2003—2007年教育振兴行动计划》的重要组成部分,精品课程建设是“质量工程”的重要内容之一。教育部计划用五年时间(2003—2007年)建设1500门国家级精品课程,利用现代化的教育信息技术手段将精品课程的相关内容上网并免费开放,以实现优质教学资源共享,提高高等学校教学质量和人才培养质量。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作,提高教学质量的若干意见》精神,紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”,在有关专家、教授的倡议和有关部门的大力支持下,我们组织并成立了“清华大学出版社教材编审委员会”(以下简称“编委会”),旨在配合教育部制定精品课程教材的出版规划,讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师,其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求,“编委会”一致认为,精品课程的建设工作从开始就要坚持高标准、严要求,处于一个比较高的起点上;精品课程教材应该能够反映各高校教学改革与课程建设的需要,要有特色风格、有创新性(新体系、新内容、新手段、新思路,教材的内容体系有较高的科学创新、技术创新和理念创新的含量)、先进性(对原有的学

科体系有实质性的改革和发展、顺应并符合新世纪教学发展的规律、代表并引领课程发展的趋势和方向)、示范性(教材所体现的课程体系具有较广泛的辐射性和示范性)和一定的前瞻性。教材由个人申报或各校推荐(通过所在高校的“编委会”成员推荐),经“编委会”认真评审,最后由清华大学出版社审定出版。

目前,针对计算机类和电子信息类相关专业成立了两个“编委会”,即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。首批推出的特色精品教材包括:

(1) 高等学校教材·计算机应用——高等学校各类专业,特别是非计算机专业的计算机应用类教材。

(2) 高等学校教材·计算机科学与技术——高等学校计算机相关专业的教材。

(3) 高等学校教材·电子信息——高等学校电子信息相关专业的教材。

(4) 高等学校教材·软件工程——高等学校软件工程相关专业的教材。

(5) 高等学校教材·信息管理与信息系统

清华大学出版社经过近 20 年的努力,在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌,为我国的高等教育事业做出了重要贡献。清华版教材经过 20 多年的精雕细刻,形成了技术准确、内容严谨的独特风格,这种风格将延续并反映在特色精品教材的建设中。

**清华大学出版社教材编审委员会**

**E-mail: dingl@tup.tsinghua.edu.cn**

# 目 录

高等学校教材·计算机科学与技术

<b>第 1 章 绪论</b> .....	1
1.1 本章知识体系结构 .....	1
1.2 典型习题解析 .....	2
1.3 教材中练习题及参考答案 .....	4
1.4 补充练习题及参考答案 .....	7
1.4.1 单项选择题 .....	7
1.4.2 填空题 .....	10
1.4.3 判断题 .....	11
1.4.4 简答题 .....	12
1.4.5 算法设计及算法分析题 .....	13
<b>第 2 章 线性表</b> .....	16
2.1 本章知识体系结构 .....	16
2.2 典型习题解析 .....	16
2.3 教材中练习题及参考答案 .....	21
2.4 补充练习题及参考答案 .....	24
2.4.1 单项选择题 .....	24
2.4.2 填空题 .....	26
2.4.3 判断题 .....	27
2.4.4 简答题 .....	28
2.4.5 算法设计题 .....	29
<b>第 3 章 栈和队列</b> .....	36
3.1 本章知识体系结构 .....	36
3.2 典型习题解析 .....	36
3.3 教材中练习题及参考答案 .....	41



3.4	补充练习题及参考答案	46
3.4.1	单项选择题	46
3.4.2	填空题	49
3.4.3	判断题	50
3.4.4	简答题	51
3.4.5	算法设计题	53
<b>第4章</b>	<b>串</b>	<b>62</b>
4.1	本章知识体系结构	62
4.2	典型习题解析	62
4.3	教材中练习题及参考答案	66
4.4	补充练习题及参考答案	67
4.4.1	单项选择题	67
4.4.2	填空题	68
4.4.3	判断题	68
4.4.4	简答题	69
4.4.5	算法设计题	70
<b>第5章</b>	<b>数组和稀疏矩阵</b>	<b>75</b>
5.1	本章知识体系结构	75
5.2	典型习题解析	75
5.3	教材中练习题及参考答案	78
5.4	补充练习题及参考答案	80
5.4.1	单项选择题	80
5.4.2	填空题	81
5.4.3	判断题	81
5.4.4	简答题	81
5.4.5	算法设计题	82
<b>第6章</b>	<b>递归</b>	<b>86</b>
6.1	本章知识体系结构	86
6.2	典型习题解析	86
6.3	教材中练习题及参考答案	88
6.4	补充练习题及参考答案	92
6.4.1	单项选择题	92
6.4.2	填空题	93
6.4.3	判断题	94
6.4.4	简答题	94

6.4.5 算法设计题 .....	95
<b>第7章 树状结构 .....</b>	<b>100</b>
7.1 本章知识体系结构 .....	100
7.2 典型习题解析 .....	101
7.3 教材中练习题及参考答案 .....	104
7.4 补充练习题及参考答案 .....	110
7.4.1 单项选择题 .....	110
7.4.2 填空题 .....	114
7.4.3 判断题 .....	116
7.4.4 简答题 .....	118
7.4.5 算法设计题 .....	128
<b>第8章 广义表 .....</b>	<b>138</b>
8.1 本章知识体系结构 .....	138
8.2 典型习题解析 .....	138
8.3 教材中练习题及参考答案 .....	140
8.4 补充练习题及参考答案 .....	142
8.4.1 单项选择题 .....	142
8.4.2 填空题 .....	143
8.4.3 判断题 .....	144
8.4.4 简答题 .....	144
8.4.5 算法设计题 .....	145
<b>第9章 图 .....</b>	<b>149</b>
9.1 本章知识体系结构 .....	149
9.2 典型习题解析 .....	150
9.3 教材中练习题及参考答案 .....	155
9.4 补充练习题及参考答案 .....	163
9.4.1 单项选择题 .....	163
9.4.2 填空题 .....	166
9.4.3 判断题 .....	168
9.4.4 简答题 .....	169
9.4.5 算法设计题 .....	179
<b>第10章 查找 .....</b>	<b>190</b>
10.1 本章知识体系结构 .....	190
10.2 典型习题解析 .....	191

10.3	教材中练习题及参考答案	194
10.4	补充练习题及参考答案	200
10.4.1	单项选择题	200
10.4.2	填空题	202
10.4.3	判断题	203
10.4.4	简答题	205
10.4.5	算法设计题	208
<b>第 11 章</b>	<b>内排序</b>	<b>212</b>
11.1	本章知识体系结构	212
11.2	典型习题解析	212
11.3	教材中练习题及参考答案	215
11.4	补充练习题及参考答案	221
11.4.1	单项选择题	221
11.4.2	填空题	223
11.4.3	判断题	224
11.4.4	简答题	224
11.4.5	算法设计题	229
<b>第 12 章</b>	<b>外排序</b>	<b>236</b>
12.1	本章知识体系结构	236
12.2	典型习题解析	236
12.3	教材中练习题及参考答案	237
12.4	补充练习题及参考答案	239
12.4.1	单项选择题	239
12.4.2	填空题	240
12.4.3	判断题	240
12.4.4	简答题	240
<b>第 13 章</b>	<b>文件</b>	<b>243</b>
13.1	本章知识体系结构	243
13.2	典型习题解析	243
13.3	教材中练习题及参考答案	244
13.4	补充练习题及参考答案	247
13.4.1	单项选择题	247
13.4.2	填空题	249
13.4.3	判断题	249
13.4.4	简答题	250

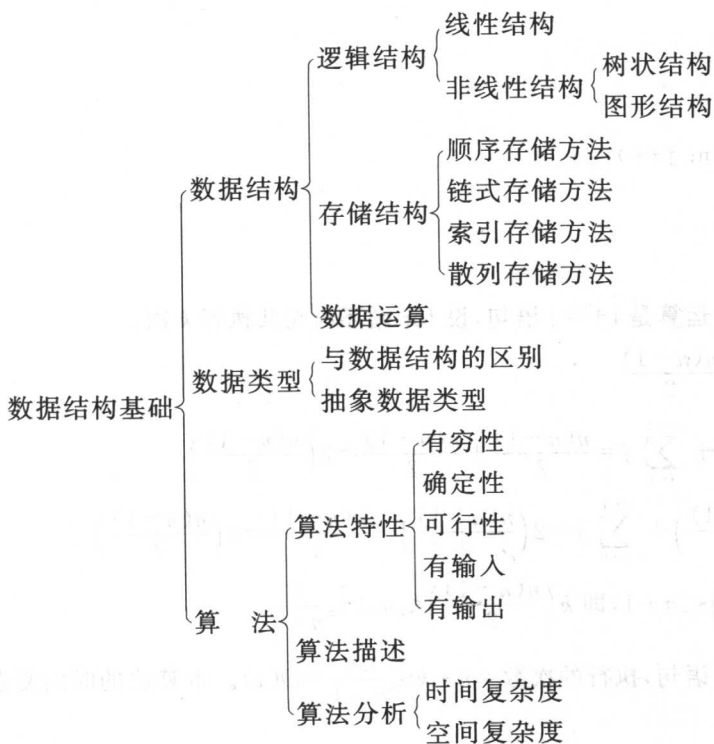
## 绪 论

**基本知识点：**数据结构和算法的概念。

**重点：**数据结构的逻辑结构、存储结构、数据运算三方面的概念及相互关系；算法时间复杂度分析。

**难点：**分析算法的时间复杂度。

## 1.1 本章知识体系结构



## 1.2 典型习题解析

**[例 1.1]** 设  $n$  为正整数,分析以下程序段中各语句的频度。

```

for (i = 0; i < n; i++)           /* 语句① */
for (j = 0; j < n; j++)         /* 语句② */
{
    c[i][j] = 0;                 /* 语句③ */
    for (k = 0; k < n; k++)     /* 语句④ */
        c[i][j] = c[i][j] + a[i][k] * b[k][j]; /* 语句⑤ */
}

```

**解:** 语句①的频度为  $n+1$  ( $i$  从 0 到  $n-1$ , 共计  $n$  次, 当  $i=n$  时还要执行一次  $i < n$  的比较, 计 1 次, 故总共  $n+1$  次); 语句②的频度为  $\sum_{i=0}^{n-1} (n+1) = n(n+1)$ ; 语句③的频度为  $\sum_{i=0}^{n-1} \sum_{j=0}^{n-1} 1 = n^2$ ; 语句④的频度为  $\sum_{i=0}^{n-1} \sum_{j=0}^{n-1} (n+1) = n^3 + n^2$ ; 语句⑤的频度为  $\sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} 1 = n^3$ 。

**[例 1.2]** 分析以下算法的时间复杂度。

```

void fun(int n)
{
    int i = 0, j;
    do
    { for (j = 0; j < n; j++)
        i += j;
    } while (i < n + 1);
}

```

**解:** 该算法中的基本运算是  $i += j$  语句, 设 for 循环语句共执行  $k$  次。

$$k=1 \text{ 时, } i = \sum_{j=0}^{n-1} j = \frac{n(n-1)}{2}$$

$$k=2 \text{ 时, } i = \frac{n(n-1)}{2} + \sum_{j=0}^{n-1} j = \frac{n(n-1)}{2} + \frac{n(n-1)}{2} = 2 \left( \frac{n(n-1)}{2} \right)$$

$$k=3 \text{ 时, } i = 2 \left( \frac{n(n-1)}{2} \right) + \sum_{j=0}^{n-1} j = 2 \left( \frac{n(n-1)}{2} \right) + \frac{n(n-1)}{2} = 3 \left( \frac{n(n-1)}{2} \right)$$

$$\text{因此, } i = k \left( \frac{n(n-1)}{2} \right) < n + 1, \text{ 即 } k \left( \frac{n(n-1)}{2} \right) \leq n, k \leq \frac{2}{n-1}$$

而基本运算是  $i += j$  语句, 执行的次数  $= n * k \leq \frac{2n}{n-1} = O(1)$ 。本算法的时间复杂度为  $O(1)$ 。

**[例 1.3]** 设  $n$  为 3 的倍数, 分析以下算法的时间复杂度。

```

void fun(int n)

```

```

{
    int i,j,x,y;
    for (i=1; i <= n; i++)
        if (3 * i <= n)
            for (j=3 * i; j <= n; j++)
                {
                    x++; y=3 * x+2;
                }
}

```

解：该算法中的基本运算是  $x++$  和  $y=3 * x+2$  语句。对于最外层的 for 循环，其执行频度为  $n+1$ ，但对于里层的 for 循环，只在  $3i \leq n$  即  $i \leq n/3$  时才执行，故基本运算的执行频度为  $\sum_{i=0}^{n/3} \sum_{j=3i}^n 1 = \sum_{i=1}^{n/3} (n - 3i + 1) = \frac{n(n-1)}{6} = O(n^2)$ 。本算法的时间复杂度为  $O(n^2)$ 。

[例 1.4] 分析以下算法的时间复杂度。

```

int fun(int m,int n)
{
    if (m==0)
        return n;
    else if (n==0)
        return m;
    else if (m>n)
        return (fun(m,n-1)+1);
    else
        return (fun(m-1,n)+1);
}

```

解：这是一个递归算法，其思路是对于输入的  $m$  和  $n$  中的较小者（不妨设为  $n$ ），一直递推减 1，直到为 0，此时返回  $m$ ，并回过来进行加 1 运算，这种递推和加 1 运算均为  $n$  次。因此，本算法的时间复杂度为  $O(\min(m,n))$ ， $\min(m,n)$  为两个数中较小者。

[例 1.5] 分析以下算法的时间复杂度。

```

void fun(ElemType A[],int i,int j,ElemType &max,ElemType &min)
{
    int mid;
    ElemType gmax,gmin,hmax,hmin;
    if (i==j)
    {
        max = min = A[i];
        return;
    }
    if (i==j-1)
    {
        if (A[i]<A[j])

```

```

    { max = A[j]; min = A[i]; }
    else
    { max = A[i]; min = A[j]; }
    return;
}
mid = (i + j) / 2;
fun(A, i, mid, gmax, gmin);
fun(A, mid + 1, j, hmax, hmin);
max = (gmax > hmax ? gmax : hmax);
min = (gmin < hmin ? gmin : hmin);
}

```

解：本算法采用递归方法求一维数组  $A[i, j]$  中的最大元素  $\max$  和最小元素  $\min$ 。本算法的主要时间花在元素比较上，设  $T(n)$  表示本算法中比较运算次数 ( $n = j - i + 1$ )，因此有以下递归表达式：

$$T(n) = \begin{cases} 0 & n = 1 (\text{即满足 } i = j \text{ 条件, 没有元素比较}) \\ 1 & n = 2 (\text{即满足 } i = j - 1 \text{ 条件, 有 1 次元素比较}) \\ T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + 2 & n > 2 (2 \text{ 次递归调用, 求 } \max \text{ 和 } \min \text{ 的 2 次元素比较}) \end{cases}$$

当  $n$  是 2 的幂时 (即存在正整数  $k$ , 使得  $n = 2^k$ ) 有：

$$\begin{aligned} T(n) &= 2 * T\left(\frac{n}{2}\right) + 2 = 2 * \left(2 * T\left(\frac{n}{4}\right) + 2\right) + 2 = 4 * T\left(\frac{n}{4}\right) + 4 + 2 = \dots \\ &= 2^{k-1} * T(2) + \sum_{i=1}^{k-1} 2^i = 2^{k-1} + 2^k - 2 = \frac{3}{2}n = O(n) \end{aligned}$$

本算法的时间复杂度为  $O(n)$ 。

## 1.3 教材中练习题及参考答案

### 1.1 简述数据与数据元素的关系与区别。

答：凡是能被计算机存储、加工的对象统称为数据，数据是一个集合。数据元素是数据的基本单位，是数据的一个元素。数据元素与数据之间的关系是元素与集合之间的关系。

### 1.2 数据结构和数据类型有什么区别？

答：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，一般包括三个方面的内容，即数据的逻辑结构、存储结构和数据的运算。而数据类型是一个值的集合和定义在这个值集上的一组运算的总称。

### 1.3 设三个函数 $f$ 、 $g$ 和 $h$ 分别为：

$$f(n) = 100n^3 + n^2 + 1000 \quad g(n) = 25n^3 + 5000n^2 \quad h(n) = n^{1.5} + 5000n \ln n$$

请判断下列关系式是否成立：

A.  $f(n) = O(g(n))$

B.  $g(n) = O(f(n))$

C.  $h(n) = O(n^{1.5})$

D.  $h(n) = O(n \ln n)$

答: A.  $f(n) = 100n^3 + n^2 + 1000 = O(n^3)$ ,  $g(n) = 25n^3 + 5000n^2 = O(n^3)$

所以,  $f(n) = O(g(n))$  成立。

B. 由 A 分析可知,  $g(n) = O(f(n))$  成立。

C. 当  $n \rightarrow \infty$  时,  $\sqrt{n} > \ln n$ , 所以  $h(n) = n^{1.5} + 5000n \ln n = O(n^{1.5})$ , 等式成立。

D. 由 C 分析可知,  $h(n) = O(n \ln n)$  等式不成立。

1.4 用 C/C++ 语言描述下列算法, 并给出算法的时间复杂度。

(1) 求一个  $n$  阶方阵的所有元素之和。

(2) 对于输入的任意三个整数, 将它们按从小到大的顺序输出。

(3) 对于输入的任意  $n$  个整数, 输出其中的最大和最小元素。

答: (1) 算法如下:

```
int sum(int A[n][n], int n)
{
    int i, j, s = 0;
    for (i = 0; i < n; i++)
        for (j = 0; j < n; j++)
            s = s + A[i][j];
    return(s);
}
```

本算法的时间复杂度为  $O(n^2)$

(2) 算法如下:

```
void order(int a, int b, int c)
{
    if (a > b)
    {
        if (b > c) printf("%d, %d, %d\n", c, b, a);
        else if (a > c) printf("%d, %d, %d\n", b, c, a);
        else printf("%d, %d, %d\n", b, a, c);
    }
    else
    {
        if (b > c)
        {
            if (a > c) printf("%d, %d, %d\n", c, a, b);
            else printf("%d, %d, %d\n", a, c, b);
        }
        else printf("%d, %d, %d\n", a, b, c);
    }
}
```

本算法的时间复杂度为  $O(1)$ 。



(3) 算法如下:

```
void maxmin(int A[], int n, int &max, int &min)
{
    int i;
    min = max = A[0];
    for (i = 1; i < n; i++)
    {
        if (A[i] > *max) max = A[i];
        if (A[i] < *min) min = A[i];
    }
}
```

本算法的时间复杂度为  $O(n)$ 。

1.5 设  $n$  为正整数, 给出下列各种算法关于  $n$  的时间复杂度。

(1)

```
void fun1(int n)
{
    i = 1, k = 100;
    while (i < n)
    {
        k = k + 1;
        i = i + 2;
    }
}
```

(2)

```
void fun2(int b[], int n)
{
    int i, j, k, x;
    for (i = 0; i < n - 1; i++)
    {
        k = i;
        for (j = i + 1; j < n; j++)
            if (b[k] > b[j]) k = j;
        x = b[i]; b[i] = b[k]; b[k] = x;
    }
}
```

(3)

```
void fun3(int n)
{
    int i = 0, s = 0;
    while (s < n)
    {
        i++;
        s = s + i;
    }
}
```

答: (1) 设 for 循环语句执行次数为  $T(n)$ , 则: