

— 市场上轻便的软考备战利器 —

软件设计师考试

考点突破、程序设计、实战练习 一本通

希赛教育软考学院 王勇 杨明

迅速抓住考点 提升解题技巧 顺利通过考试
**阅卷专家
经验钜献**

经典畅销软考图书专家作者团队新品再续

- 恰如其分的考点分析，不多不少，不深不浅
- 典型常考的试题分析，深度、广度、方法、技巧，一应俱全
- 近似考题的实战练习，适时、量多、逼真
- 细致周到的练习剖析，全面、突出、巩固



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

— 市场上轻便的软考备战利器 —

软件设计师考试

考点突破、程序设计、实战练习 一本通

希赛教育软考学院 王 勇 杨 明 主编

電子工業出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书由希赛教育软考学院组编，作为计算机技术与软件专业技术资格（水平）考试中的软件设计师级别的考试辅导指定教材。内容紧扣考试大纲，通过对历年试题进行科学分析、研究、总结、提炼而成。每章内容分为考点突破、典型试题分析、实战练习和练习题解析 4 个部分。

基于历年试题，利用统计分析的方法，科学作出结论并预测以后的出题动向，是本书的一大特色。本书可以保证既不漏掉考试必需的知识点，又不加重考生备考负担，使考生轻松、愉快地掌握知识点并领悟软件设计师考试的真谛。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目（CIP）数据

软件设计师考试考点突破、程序设计、实战练习一本通 / 王勇，杨明主编.
北京：电子工业出版社，2010.10
ISBN 978-7-121-11754-1

I. ①软… II. ①王… ②杨… III. ①软件设计—工程技术人员—资格考核—自学参考资料
IV. ①TP311.5

中国版本图书馆 CIP 数据核字（2010）第 172962 号

责任编辑：孙学瑛

印 刷：北京天宇星印刷厂

装 订：三河市鹏成印业有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：787×1092 1/16 印张：37.5 字数：944 千字

印 次：2010 年 10 月第 1 次印刷

印 数：4000 册 定价：69.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

前言

全国计算机技术与软件专业技术资格（水平）考试（俗称“软考”）由人事部、工业和信息化部主办，面向社会，用于考查计算机专业人员的水平与能力。考试客观、公正，得到了社会的广泛认可，并实现了中、日、韩三国互认。

本书内容紧扣考试大纲，采用了表格统计法、走势图分析法，科学地研究每个知识点的命题情况，准确把握每个出题点的深浅，同时基于每个章节知识点分布统计分析的结果，科学地编写实战练习题，完全紧扣大纲，结构科学、重点突出、针对性强。

内容超值，针对性强

本书每章的内容分为考点突破、典型试题分析、实战练习题和练习题解析4个部分。

第一部分为考点突破。该部分首先对历年试题进行了统计分析，采用图表达形式，形象而直观，使各考点“暴露无遗”，通过学习本部分内容，考生可以对考试的知识点分布、考试重点有一个整体上的认识和把握；然后重点对大纲中所规定的重要考试内容和考试必备的知识点进行了“画龙点睛”，章节中的知识点解析深浅程度根据该知识点在历年试题中的统计分析结果而定。

第二部分为典型试题分析。该部分从历年考试真题中抽取具有代表性的、经常考到的试题进行详细的分析，通过阅读这一部分，考生可以熟悉考试方法、试题形式，试题的深度和广度，以及内容的分布、解答问题的方法和技巧。

第三部分为实战练习题。该部分给出了多道试题，根据考点突破部分的知识点统计、分析的结果而命题。这些试题与考试真题具有很大的相似性，用来检查考生学习前面两个部分内容的效果。

第四部分为练习题解析。该部分是实战练习部分的补充，为实战练习的所有习题进行了较详细的分析，并给出了解答。考生需要掌握每个练习题及其解答，这一部分可以帮助考生温习和巩固前面所学的知识，这种辅导方式保证内容全面，突出重点，为考生打造一条通向考试终点的捷径。

作者权威，阵容强大

希赛教育（www.educity.cn）专业从事人才培养、教育产品开发、教育图书出版，在职业教育方面具有极高的权威性。特别是在在线教育方面稳居国内首位，希赛教育

的远程教育模式得到了国家教育部门的认可和推广。

希赛教育软考学院(www.csairk.com)是全国计算机技术与软件专业技术资格(水平)考试的顶级培训机构,拥有近 20 名资深软考辅导专家,负责高级资格的考试大纲制定以及软考辅导教材的编写工作,共组织编写和出版了 60 多本软考教材,内容涵盖了初级、中级和高级的各个专业,包括教程系列、辅导系列、考点分析系列、冲刺系列、串讲系列、试题精解系列、疑难解答系列、全程指导系列、案例分析系列、指定参考用书系列和一本通 11 个系列的书籍。希赛教育软考学院的专家录制了软考培训视频教程、串讲视频教程、试题讲解视频教程和专题讲解视频教程 4 个系列的软考视频,希赛教育软考学院的软考教材、软考视频、软考辅导为考生助考、提高通过率做出了不可磨灭的贡献,在软考领域有口皆碑。特别是在高级资格领域,无论是考试教材,还是在线辅导和面授,希赛教育软考学院都独占鳌头。

本书由希赛教育软考学院王勇和杨明组织编写,参加编写工作的人员还有张友生、施游、朱小平、刘毅、李雄、胡钊源和桂阳等,另外,何玉云和周玲参加了校对和审核工作。

在线测试,心中有数

上学吧(www.shangxueba.com)在线测试平台为考生准备了在线测试,其中有数十套全真模拟试题和考前密卷,考生可选择任何一套进行测试。测试完毕,系统自动判卷,立即给出分数。

对于考生做错的地方,系统会自动记忆,待考生第二次参加测试时,可选择“试题复习”。这样,系统就会自动把考生原来做错的试题显示出来,供考生重新测试,以加强记忆。

如此,读者可利用上学吧在线测试平台的在线测试系统检查自己的实际水平,加强考前训练,做到心中有数,考试不慌。

诸多帮助,诚挚致谢

在本书出版之际,要特别感谢全国软考办的命题专家们,编者在本书中引用了部分考试原题,使本书能够尽量方便读者的阅读。在本书的编写过程中,参考了许多相关的文献和书籍,编者在此对这些参考文献的作者表示感谢。

感谢电子工业出版社孙学瑛老师,她在本书的策划、选题的申报、写作大纲的确定,以及编辑、出版等方面,付出了辛勤的劳动和智慧,给予了我们很多的支持和帮助。

感谢参加希赛教育软考学院辅导和培训的学员,正是他们的想法汇成了本书的源动力,他们的意见使本书更加贴近读者。

考点突破、程序设计、实战练习一本通

由于编者水平有限，且本书涉及的内容较广，书中难免存在错漏和不妥之处，编者诚恳地期望各位专家和读者不吝指正和帮助，对此，我们将十分感激。

互动讨论，专家答疑

希赛教育软考学院（www.csairk.com）是中国最大的软考在线教育网站，该网站论坛是国内人气最旺的软考社区，在这里，读者可以和数十万考生进行在线交流，讨论有关学习和考试的问题。希赛教育软考学院拥有强大的师资队伍，为读者提供全程的答疑服务，在线回答读者的提问。

有关本书的意见反馈和咨询，读者可在希赛教育软考学院论坛“软考教材”版块中的“希赛教育软考学院”栏目上与作者进行交流。

希赛教育软考学院

2010年8月

目 录

第1章 计算机硬件基础知识1

- 1.1 考点突破1
 - 1.1.1 历年考试情况分析2
 - 1.1.2 数据的表示3
 - 1.1.3 计算机系统的组成与体系结构5
 - 1.1.4 存储系统10
 - 1.1.5 可靠性与系统性能评测13
- 1.2 典型试题分析19
- 1.3 实战练习题24
- 1.4 练习题解析26

第2章 操作系统基础知识31

- 2.1 考点突破31
 - 2.1.1 历年考试情况分析31
 - 2.1.2 进程管理33
 - 2.1.3 存储管理37
 - 2.1.4 磁盘管理39
 - 2.1.5 设备管理40
 - 2.1.6 文件管理42
- 2.2 典型试题分析44
- 2.3 实战练习题52
- 2.4 练习题解析54

第3章 程序语言和语言处理程序基础知识58

- 3.1 考点突破58
 - 3.1.1 历年考试情况分析58

- 3.1.2 汇编、编译、解释系统基础 60

- 3.1.3 程序设计语言基础 67
- 3.1.4 各种程序语言的特点 70
- 3.2 典型试题分析 72
- 3.3 实战练习题 75
- 3.4 练习题解析 77

第4章 数据结构与算法基础 82

- 4.1 考点突破 82
 - 4.1.1 历年考试情况分析 82
 - 4.1.2 数据结构基础与线性表 84
 - 4.1.3 广义表 90
 - 4.1.4 树与二叉树 90
 - 4.1.5 图 94
 - 4.1.6 算法基础 96
 - 4.1.7 排序与查找 100
- 4.2 典型试题分析 108
- 4.3 实战练习题 121
- 4.4 练习题解析 126

第5章 数据库系统基础知识 137

- 5.1 考点突破 137
 - 5.1.1 历年考试情况分析 137
 - 5.1.2 数据库模式 139
 - 5.1.3 ER 模型 141
 - 5.1.4 关系代数与元组演算 142
 - 5.1.5 规范化理论 145
 - 5.1.6 SQL 语言基础 150

5.1.7 并发控制	154
5.2 典型试题分析	158
5.3 实战练习题	163
5.4 练习题解析	167

第6章 网络基础知识

6.1 考点突破	174
6.1.1 历年考试情况分析	174
6.1.2 开放系统互联参考模型	175
6.1.3 TCP/IP 协议族	179
6.1.4 IP 地址与子网划分	182
6.1.5 网络规划与设计	185
6.1.6 网络应用	186
6.2 典型试题分析	190
6.3 实战练习题	195
6.4 练习题解析	197

第7章 软件工程基础知识

7.1 考点突破	204
7.1.1 历年考试情况分析	205
7.1.2 软件生命周期与软件开发模型	207
7.1.3 系统开发方法论	216
7.1.4 软件测试	218
7.1.5 软件维护	225
7.1.6 软件质量保证	226
7.1.7 软件过程改进	229
7.1.8 软件项目管理	231
7.1.9 软件开发工具	235
7.2 典型试题分析	237
7.3 实战练习题	241
7.4 练习题解析	245

第8章 面向对象基础知识

8.1 考点突破	254
----------------	-----

8.1.1 历年考试情况分析	254
8.1.2 面向对象的基本概念	256
8.1.3 面向对象分析与设计基础	258
8.1.4 UML 基础	260
8.1.5 设计模式	262
8.2 典型试题分析	278
8.3 实战练习题	281
8.4 练习题解析	283

第9章 信息安全知识

9.1 考点突破	287
9.1.1 历年考试情况分析	287
9.1.2 安全基础技术	288
9.1.3 网络安全	290
9.1.4 计算机病毒与木马	296
9.2 典型试题分析	298
9.3 实战练习题	299
9.4 练习题解析	300

第10章 多媒体基础知识

10.1 考点突破	302
10.1.1 历年考试情况分析	302
10.1.2 多媒体基础概念	303
10.1.3 多媒体相关计算	305
10.1.4 常见多媒体标准	306
10.2 典型试题分析	312
10.3 实战练习题	318
10.4 练习题解析	319

第11章 知识产权与标准化基础知识

11.1 考点突破	324
11.1.1 历年考试情况分析	324
11.1.2 知识产权	325

11.1.3 标准化法	327	14.1.6 活动图	441
11.2 典型试题分析	329	14.1.7 状态图	445
11.3 实战练习题	333	14.1.8 通信图	447
11.4 练习题解析	334	14.1.9 构件图	447
第 12 章 数据流图技术	339	14.1.10 部署图	448
12.1 考点突破	339	14.2 典型试题分析	449
12.1.1 历年考试情况分析	339	14.3 实战练习题	460
12.1.2 数据流图基本概念	340	14.4 练习题解析	471
12.1.3 数据字典	341	第 15 章 数据结构与算法设计	482
12.1.4 数据平衡原则	342	15.1 考点突破	482
12.1.5 答题技巧	343	15.1.1 历年考试情况分析	482
12.2 典型试题分析	344	15.1.2 递归法	483
12.3 实战练习题	359	15.1.3 分治法	485
12.4 练习题解析	371	15.1.4 动态规划法	486
第 13 章 数据库建模与 SQL 技术	384	15.1.5 贪心法	487
13.1 考点突破	384	15.1.6 回溯法	495
13.1.1 历年考试情况分析	384	15.2 典型试题分析	498
13.1.2 数据库设计基础	385	15.3 实战练习题	513
13.1.3 概念设计	386	15.4 练习题解析	523
13.2 典型试题分析	390	第 16 章 面向对象程序设计	530
13.3 实战练习题	405	16.1 考点突破	530
13.4 练习题解析	415	16.1.1 历年考试情况分析	530
第 14 章 UML 建模技术	427	16.1.2 C++ 语言语法要点	531
14.1 考点突破	427	16.1.3 Java 语言语法要点	539
14.1.1 历年考试情况分析	427	16.2 典型试题分析	542
14.1.2 概述	428	16.3 实战练习题	561
14.1.3 用例图	428	16.4 练习题解析	581
14.1.4 类图与对象图	433	主要参考文献	590
14.1.5 顺序图	441		

1

第 1 章

计算机硬件基础知识

根据考试大纲，本章要求考生掌握以下几个方面的知识点。

(1) 数据的表示：数制及其转换、原码、反码、补码、移码、浮点数运算、溢出、算术运算、逻辑运算、校验码。

(2) 计算机系统的组成、体系结构分类及特性：CPU、存储器的组成、性能和基本工作原理、常用 I/O 设备、通信设备的性能及基本工作原理、I/O 接口的功能、类型和特性、CISC/RISC、流水线操作、多处理机、并行处理。

(3) 存储系统：虚拟存储器基本工作原理、多级存储体系、RAID 类型和特性。

(4) 可靠性与系统性能评测基础知识：诊断与容错、系统可靠性分析评价、校验方法、计算机系统性能评测方法。

1.1 考点突破

从历年的考试情况来看，本章的考点主要集中在以下几个方面。

- 在数据的表示中，主要考查浮点数运算、溢出、算术运算、逻辑运算。
- 在计算机系统的组成与体系结构中，主要考查计算机体系结构分类、指令系统基础、CISC 与 RISC、流水线操作等相关内容。
- 在存储系统中，主要考查 Cache 存储器。
- 在可靠性与系统性能评测基础知识中，主要考查系统可靠性分析和校验方法。

1.1.1 历年考试情况分析

在历年的考试试题中,有关计算机硬件基础知识的试题如表 1-1 所示。

表 1-1 计算机硬件基础知识的试题分布表

题号	2006.05	2006.11	2007.05
1	溢出	内存容量的计算	控制器的组成
2	浮点数的运算	串并联系统的可靠度计算	Cache
3	Cache	流水线的执行时间	RISC 指令系统的设计原则
4		计算机指令平均时钟数的计算	串并联系统的可靠度计算
5	平均故障间隔时间	计算机运算速度的计算	流水线的执行时间
6	流水线的吞吐率计算	扩展操作码的设计	系统响应时间与作业吞吐率
题号	2007.11	2008.05	2008.11
1	立即寻址方式	计算机基本工作原理	计算机内存布局与指令寻址方式
2	直接寻址方式	流水线的操作周期	浮点数运算特点
3	系统响应时间与作业吞吐率	内存容量的计算	寄存器的类型与特点
4	流水线的执行时间	数据总线	I/O 设备与主机的数据交换
5		计算机体系结构分类	校验方法的比较
6	内存容量的计算	段式存储管理的优点	Cache
32		可靠性计算	
34	可靠性计算		
题号	2009.05	2009.11	2010.05
1	校验方法	CPU 的组成与工作原理	CPU 的组成
2	磁盘容量	CISC 与 RISC 的区别	串并联系统的可靠度计算
3	存储器的分类	浮点数运算特点	中断的概念
4	总线的分类		算术、逻辑运算
5	数据的表示	校验方法	计算机指令结构
6	流水线的吞吐率计算	Cache	CPU 的性能指标
20			数制转换

按照知识点进行总结和归类的试题分布情况如表 1-2 所示。

表 1-2 计算机硬件基础知识归类表

知识点	2006.05	2006.11	2007.05	2007.11	2008.05	2008.11	2009.05	2009.11	2010.05
数据的表示	2	0	0	0	0	1	1	2	2
计算机系统的组成与体系结构	1	2	4	5	4	2	2	2	4
存储系统	2	1	1	1	2	2	2	1	0
可靠性与系统性能评测	1	3	1	0	0	1	1	1	1
合计	6	6	7	7	6	6	6	6	7

从表 1-2 中可以看出,计算机硬件基础知识方面的内容在历年的考试中分值非常稳定,一直维持在 6~7 分。其所占分数比例的趋势如图 1-1 所示。

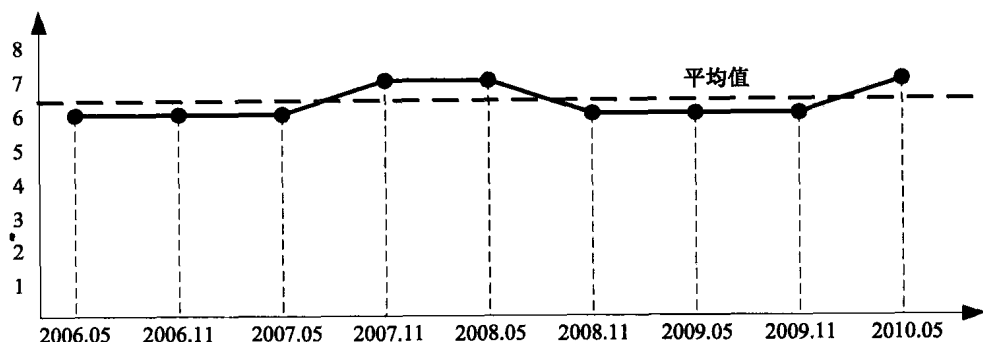


图 1-1 计算机硬件基础知识历年试题比例趋势图

1.1.2 数据的表示

数据的表示部分包含了数据转换、原码、反码、补码、移码及浮点数运算等知识。其中难点是浮点数运算。

1. 数制转换

(1) R 进制数转换成十进制数

R 进制数转换成十进制数通常使用按权展开法。具体操作方式为：将 R 进制数的每一位数值用 R^k 形式表示，即幂的底数是 R ，指数为 k ， k 与该位和小数点之间的距离有关。当该位位于小数点左边时， k 值是该位和小数点之间数码的个数，而当该位位于小数点右边时， k 值是负值，其绝对值是该位和小数点之间数码的个数加 1。

例如，二进制数 10100.01 的值可计算如下：

$$10100.01 = 1 \times 2^4 + 1 \times 2^2 + 1 \times 2^{-2}$$

按照上面的表示法，即可计算出 R 进制数十进制的值。

(2) 十进制数转换为 R 进制数

最常用的是“除以 R 取余法”。例如将十进制数 94 转换为二进制数：

2 94	余	0
2 47		1
2 23		1
2 11		1
2 5		1
2 2		0
1		1

将所得的余数从低位到高位排列 $(1011110)_2$ 就是 94 的二进制数。

(3) 二进制数与八进制数、十六进制数之间的转换

- 二进制转八进制：将每 3 位二进制数转换为八进制数。
- 二进制转十六进制：将每 4 位二进制数转换为八进制数。
- 八进制转二进制：将每个八进制数转换为 3 位二进制数。
- 十六进制转二进制：将每个十六进制数转换为 4 位二进制数。

上面的转换都是以小数点作为计算数码个数的起点。八进制数和十六进制数转换可先转换为二进制数，然后再转换为目标进制。

2. 原码、反码、补码、移码

在计算机中，数据编码方式可以有多种，最为常见的有原码、反码、补码、移码。一个正数的原码、补码、反码是相同的，负数则不同。

(1) 原码

将最高位用作符号位（0 表示正数，1 表示负数），其余各位代表数值本身的绝对值的表示形式。这种方式是最容易理解的。

例如，+1 的原码是 0000 0001，-1 的原码是 1000 0001。

但是直接使用原码在计算时却会有麻烦，例如 $(1)_{10} + (-1)_{10} = 0$ ，如果直接使用原码，则：

$$(0000\ 0001)_2 + (1000\ 0001)_2 = (1000\ 0010)_2$$

这样计算的结果是-2，也就是说，使用原码直接参与计算可能会出现错误的结果。所以，原码的符号位不能直接参与计算，必须和其他位分开，这样会增加硬件的开销和复杂性。

(2) 反码

正数的反码与原码相同。负数的反码符号位为 1，其余各位为该数绝对值的原码按位取反。这个取反的过程使得这种编码称为“反码”。

例如，-1 的反码：1111 1110。

同样对上面的加法，使用反码的结果是：

$$(0000\ 0001)_2 + (1111\ 1110)_2 = (1111\ 1111)_2$$

这样的结果是负 0，而在人们普遍的观念中，0 是不分正负的。反码的符号位可以直接参与计算，而且减法也可以转换为加法计算。

(3) 补码

正数的补码与原码相同。负数的补码是该数的反码加 1，这个加 1 就是“补”。

例如，-1 的补码：1111 1110+1 = 1111 1111。

再次做加法是这样的：

$$(0000\ 0001)_2 + (1111\ 1111)_2 = (0000\ 0000)_2$$

直接使用补码进行计算的结果是正确的。

对一个补码表示的数，要计算其原码，只要对它再次求补，即可得到该数的原码。

由于补码能使符号位与有效值部分一起参加运算，从而简化运算规则，同时它也使减法运算转换为加法运算，进一步简化计算机中运算器的电路，这使得在大部分计算机系统中，数据都使用补码表示。

(4) 移码

移码是对补码的符号位取反得到的一种编码，它只用于表示浮点数的阶码，所以只用于整数。

例如，-1 的移码为：0111 1111。

3. 浮点数运算

在数学中，要表示一个很大的数时，我们常常使用一种称为科学计数法的方式：

$$N = M * R^e$$

其中 M 称为尾数， e 是指数， R 为基数。

浮点数就是使用这种方法来表示大范围的数，其中指数一般是 2、8、16，而且对于特定机器而言，指数是固定不变的，所以在浮点数中指数并不出现。从这个表达式可以看出：浮点数表示的精读取决于尾数的宽度，范围取决于基数的大小和指数的宽度。

浮点数的运算主要有 3 个步骤：对阶、尾数计数、结果格式化。

(1) 对阶

首先计算两个数的指数差，把指数小的向指数大的对齐，并将尾数右移指数差的位数，这样两个浮点数就完成了对阶的操作。可以看出，对阶的过程可能使得指数小的浮点数失去一些有效位。如果两个浮点数阶数相差很大，大于指数小的浮点数的尾数宽度，那么对阶后那个浮点数的尾数就变成了 0，即当作机器零处理了。

(2) 尾数计算

对阶完成后，两个浮点数尾数就如同定点数，计算过程同定点数计算。

(3) 结果格式化

尾数计算后，可能会产生溢出，此时将尾数右移，同时指数加 1，如果指数加 1 后发生了溢出，则表示两个浮点数的运算发生了溢出。

如果尾数计算没有溢出，则尾数不断左移，同时指数减 1，直到尾数为格式化数。如果在这个过程中，指数小于机器能表达的最小数，则将结果置“机器零”，这种情况称为下溢。

1.1.3 计算机系统的组成与体系结构

在计算机系统的组成与体系结构中，计算机体系结构分类、指令系统基础、CISC 与 RISC、流水线操作等内容十分重要，下面将详细介绍这几个方面的知识。

1. 计算机体系结构分类

计算机体系结构分类有多种方式，其中最为常见的是：Flynn 分类法与冯氏分类法。而考试中主要考查的是 Flynn 分类法。

Flynn 分类法是根据指令流、数据流和多倍性 3 方面来进行分类的，如表 1-3 所示。

表 1-3 Flynn 分类法

体系结构类型	结 构	关键特性	代 表
单指令流单数据流 SISD	控制部分：一个 处 理 器：一个 主存模块：一个		单处理器系统
单指令流多数据流 SIMD	控制部分：一个 处 理 器：多个 主存模块：多个	各处理器以异步的形式执行同一条指令	并行处理机 阵列处理机 超级向量处理机
多指令流单数据流 MISD	控制部分：多个 处 理 器：多个 主存模块：多个	被证明不可能，至少是不实际	理论模型，未见成形产品
多指令流多数据流 MIMD	控制部分：多个 处 理 器：多个 主存模块：多个	能够实现作业、任务、指令等各级全面并行	多处理机系统 多计算机

2. 计算机的硬件组成

计算机硬件系统由运算器、控制器、存储器、输入设备和输出设备 5 大部件组成。其中，运算器和控制器组成中央处理器（CPU）。运算器负责完成算术、逻辑运算功能，通常由 ALU（算术/逻辑单元）、寄存器、多路转换器、数据总线组成；控制器则负责依次访问程序指令，进行指令译码，并协调其他设备，通常由程序计数器（PC）、指令寄存器、指令译码器、状态/条件寄存器、时序发生器、微操作信号发生器组成。

3. 指令系统基础

在计算机中，CPU 都会定义出自己特定的指令系统，不过都遵循着统一的标准格式。指令的基本格式是由操作码和地址码两个部分组成的。操作码指出该指令要完成什么操作，地址码则是提供原始的数据。指令系统中定义操作码的方式可以分为规整型（定长编码）和非规整型（变长编码）两种，如表 1-4 所示。

表 1-4 指令系统操作码定义分类方法比较表

编码方式	编码方式	平均码长
定长编码	采用相等码长，每个操作码的长度相等	码长 $\geq \log_2$ （操作码数） 如 14 个操作码，就应该是 4 位： $2^3=8$ ，不够， $2^4=16$ ，多两个
变长编码	根据使用频度选择不同长度的编码	将码分为几类，然后再对每类进行编码 平均码长为：将每个码长乘以频度，再累加其和

而在指令系统中用来确定如何提供操作数或提供操作数地址的方式称为寻址方式和编址方式。操作数可以存放在 CPU 中的寄存器（用寄存器名操作）、主存储器（指出存储单元地址）、堆栈（先进后出的存储机制，用栈顶指针 SP 来标出其当前位置）、

外存储器或外围设备中。不过在运算时，数据均在主存储器中，操作数可以采用以下几种寻址方式：

(1) 隐含寻址方式

在指令中不明显地给出而是隐含着操作数的地址。例如，单地址的指令格式，没有在地址字段中指明第二操作数地址，而是规定累加寄存器 AC 作为第二操作数地址，AC 对单地址指令格式来说是隐含地址。

(2) 立即寻址方式

指令的地址字段指出的不是操作数的地址，而是操作数本身。这种方式的特点是指令执行时间很短，不需要访问内存取数。题目中所说的“操作数包含在指令中的寻址方式”就是立即寻址。

例如，单地址的移位指令格式为

OP (移位)	F	D
---------	---	---

这里 D 不是地址，而是一个操作数。F 为标志位，当 F=1 时，操作数进行右移；当 F=0 时，操作数进行左移。

(3) 直接寻址方式

直接寻址的特点是：在指令格式的地址字段中直接指出操作数在内存的地址 D。

在采用直接寻址方式时，指令字中的形式地址 D 就是操作数的有效地址 E，即 $E=D$ 。因此，通常把形式地址 D 又称为直接地址。此时，由寻址模式给予指示。如果用 S 表示操作数，那么直接寻址的逻辑表达式为 $S=(E)=(D)$ 。

(4) 间接寻址方式

在间接寻址的情况下，指令地址字段中的形式地址 D 不是操作数的真正地址，而是操作数地址的指示器，D 单元的内容才是操作数的有效地址。

如果把直接寻址和间接寻址结合起来，指令有如下形式：

操作码	I	D
-----	---	---

寻址特征位 $I=0$ ，表示直接寻址，这时有效地址 $E=D$ ； $I=1$ ，表示间接寻址，这时有效地址 $E=(D)$ 。

间接寻址方式是早期计算机中经常采用的方式，但由于两次访问内存，影响指令执行速度，现在已很少使用。

(5) 寄存器寻址方式和寄存器间接寻址方式

当操作数不放在内存中，而是放在 CPU 的通用寄存器中时，可采用寄存器寻址方式。此时指令中给出的操作数地址不是内存的地址单元号，而是通用寄存器的编号。这也就是题目中所说的“操作数在寄存器中的寻址方式”。

寄存器间接寻址方式与寄存器寻址方式的区别在于：指令格式中的寄存器内容不是操作数，而是操作数的地址，该地址指明的操作数在内存中。这也就是题目中所说的“操作数的地址在寄存器中的寻址方式”。

(6) 相对寻址方式

相对寻址方式是把程序计数器 PC 的内容加上指令格式中的形式地址 D 而形成操作数的有效地址。程序计数器的内容就是当前指令的地址。“相对”寻址就是相对于当前的指令地址而言的。

采用相对寻址方式的优点是程序员无须用指令的绝对地址编程，且所编程序可以放在内存中的任何位置。

此时形式地址 D 通常称为偏移量，其值可正可负，相对于当前指令地址进行浮动。

(7) 基址寻址方式

基址寻址方式是将 CPU 中基址寄存器的内容加上指令格式中的形式地址而形成操作数的有效地址。它的优点是可以扩大寻址能力。与形式地址相比，基址寄存器的位数可以设置得很长，从而可以在较大的存储空间中寻址。

(8) 变址寻址方式

变址寻址方式与基址寻址方式计算有效地址的方法很相似，它把 CPU 中某个变址寄存器的内容与偏移量 D 相加来形成操作数有效地址。但使用变址寻址方式的目的在于扩大寻址空间，而在于实现程序块的规律性变化。

4. CISC 与 RISC

为了提高操作系统的效率，人们最初选择了向指令系统中添加更多、更复杂的指令，而随着不断升级和向后兼容的需要，指令集也越来越大。这种类型的计算机，我们称之为复杂指令计算机 CISC。而后来研究发现，计算机指令系统如果使用少量结构简单的指令会提高计算机的性能，这就是精简指令集计算机 RISC。计算机执行程序所需的时间 P 由 3 方面因素决定：编译后产生的机器指令数 I 、执行每条指令所需的平均周期数 CPI 以及每个机器周期的时间 T 。它们的关系是 $P=I \times CPI \times T$ 。RISC 正是通过简化指令的途径使计算机结构更合理，减少指令执行周期数，提高运算速度。虽然 RISC 编译后产生的机器指令数 (I) 增多了，但指令所需的周期数 (CPI) 和每个周期的时间 (T) 都可以减少。它与 CISC 可谓各有特色，如表 1-5 所示。

表 1-5 指令系统操作码定义分类方法比较表

指令系统类型	指令	寻址方式	实现方式	其他
CISC (复杂)	数量多，使用频率差别大，可变长格式	支持多种	微程序控制技术	
RISC (精简)	数量少，使用频率接近，定长格式，大部分为单周期指令，操作寄存器，只有 Load/Store 操作内存	支持方式少	增加了通用寄存器；硬布线逻辑控制为主；采用流水线	优化编译，有效支持高级语言

典型的 RISC 处理器包括：DEC 的 Alpha 21164、IBM 的 Power PC620、HP 的 PA-8000、SGI MIPS 分部的 TS、Sun 的 Ultra SPARC。目前 RISC 处理器技术的发展方向是采用并行处理技术（包括超级流水线、超级标量、超长指令字）大幅度提高运算速度。

5. 流水线

流水线是指在程序执行时多条指令重叠进行操作的一种准并行处理实现技术。各