

全国计算机应用软件人员水平考试辅导丛书

软件人员水平考试 题例和选解

(程序员级 高级程序员级)

第 二 版

杨德元 朱慧真 主编

清华大学出版社

TP311.56
1250

全国计算机应用软件人员水平考试辅导丛书

软件人员水平考试 题例和选解

(程序员级 高级程序员级)

第二版

杨德元 朱慧真 主编

清华大学出版社

内 容 简 介

本书为计算机软件专业技术资格和水平考试的题例和选解,帮助软件人员复习有关考试内容,熟悉并掌握考试典型题例的分析和解题方法。它是在中国软件行业协会考试指导中心组织下,由北京大学、清华大学等院校的有关专家,在第一版的基础上,依照1991年新公布的大纲修订、编写而成的。内容包括:基础知识、程序设计及近期国内软件水平考试的典型试题的分析、解法,并附有答案。全书题例共200多道,系统、全面、典型。

本书可供程序员级、高级程序员级软件人员学习,也可作大专院校师生的教学参考书,或供各部门举办辅导班选用。本书也是中央电视台电视辅导讲座的参考读物。

全国计算机应用软件人员水平考试辅导丛书

软件人员水平考试题例和选解

(程序员级 高级程序员级)

第二版

杨德元 朱慧真 主编

☆

清华大学出版社出版

北京 清华园

北京市昌平环球科技印刷厂印刷

新华书店总店科技发行所发行

☆

开本: 787×1092 1/16 印张: 22.5 字数: 543千字

1991年6月第2版 1991年6月第1次印刷

印数: 00001~35000

ISBN 7-302-00889-2/TP·323

定价: 8.60元

《软件人员水平考试题例和选解》编委会

顾问 杨天行 徐家福 杨芙清 王尔乾

主编 杨德元 朱慧真

编委 (按姓氏笔划为序)

王爱英 史美林 卢义明 刘椿年 刘玉民

苏民生 沈林兴 李 晨 罗晓沛 屈婉玲

杨冬青 陈向群 严忠琛 孟庆昌 林钧礼

张 治 杨自强 郑人杰 赵凤治 唐世渭

徐国平 耿素云 耿继秀 柳纯录 韩 奕

唐肖光 殷志鹤 殷树勋 管有庆 潘锦平

前 言

计算机软件人员水平考试是造就宏大的多层次计算机人才队伍的一项重要措施，从1991年起，在全国实施《中国计算机软件专业技术资格和水平考试暂行规定》，考试级别分为初级程序员（相当于技术员）级、程序员（相当于助理工程师）级、高级程序员（相当于工程师）级、系统分析员（相当于高级工程师）级。

为了指导软件人员的学习，帮助准备参加软件专业技术资格和水平考试的学员应试，我们在原编写出版的《全国计算机应用软件人员水平考试辅导丛书》基础上，重新聘请了北京大学、清华大学、中国科大研究生院、上海交通大学、南京大学、南京东南大学等单位的专家组成编委会，根据1991年3月公布的考试大纲，进一步修订成本丛书。

本丛书共包括：《程序员水平考试指导》、《高级程序员水平考试指导》、《程序设计考试指导》、《软件人员水平考试题例和选解》及《系统分析员教程》。

必须说明，本丛书是供考试应考人员参考的综合性辅导教材，供读者在自学基础知识后复习提高用，它的主要特点是全面（覆盖了考试大纲的主要内容）、系统（而不是给出零碎的知识），并且体现了基本要求（包括基本概念，基本方法和基本原理）。限于篇幅，各章节不可能全面展开。为解决读者自学问题，我们将另行组织编写《全国计算机软件人员水平考试系列教材》。

本丛书亦可用作在职软件人员进修教材，或计算机应用专业师生的参考书籍，本丛书也是中央电视台播放的软件人员水平考试电视辅导讲座的教材。

本书供程序员与高级程序员级使用，内容包括基础知识和程序设计，在第二版中增加了C语言程序设计、汇编语言程序设计及1989—1990年水平考试典型试题分析。本书由杨德元、朱慧真主编，参加修订的有：朱慧真、罗晓沛、唐世渭、耿素云、屈婉玲、杨冬青、孟庆昌、苏民生、韩奕、刘玉民、唐肖光、管有庆等同志，徐国平参加编审。

在本丛书的编写过程中，曾得到中国计算机学会的指导和帮助。清华大学出版社为使本丛书迅速出版付出了辛勤劳动，在此一并致谢！

中国软件行业协会考试指导中心

1991年4月

目 录

第一部分 基础知识题例和选解

第一章 硬件基础知识	2
第二章 软件技术基础	31
第三章 应用基础	34

第二部分 程序设计题例和选解

第四章 程序设计基础	80
第五章 FORTRAN 程序设计	93
第六章 PASCAL 程序设计	130
第七章 COBOL 程序设计	177
第八章 C 程序设计	237
第九章 CASL 程序设计	253

第三部分 软件人员水平考试试题解析

第十章 1989、1990年全国软件人员水平考试典型试题分析	276
--------------------------------------	-----

附录 I CASL 汇编语言简介	346
------------------------	-----

附录 II ASCII 字符与编码对照表	351
----------------------------	-----

第 一 部 分

基础知识题例和选解

第一章 硬件基础知识

1 从供选择的答案中选出正确的答案填入 内。把十进制数 205.5 转化为二进制数为 a , 转化为八进制数为 b , 转化为十六进制数为 c 。

〔供选择的答案〕

a : ①11001101.1; ②11010101.01; ③11001011.1; ④01100101.01。

b : ①315.4; ②315.1; ③325.2; ④315.4; ⑤245.1。

c : ①CD.1; ②CD.8; ③65.1; ④CB.8; ⑤D5.1。

〔答案〕

a —①; b —④; c —②。

〔分析〕

首先把十进制数 205.5 分别按整数部分和小数部分转化为二进制数。205 化成二进制数为 11001101, 小数部分 0.5 化为二进制数 0.1。所以 205.5 用二进制数表示为 11001101.1。

再以二进制数为基础化成八进制数。方法是以二进制数的小数点为界, 整数部分向左, 小数部分向右, 每三位二进制数为一组, 不足三位的用“0”补足, 然后用八进制数书写。

本题中, 整数部分每三位一组为 11001101, 写成八进制数为 315。小数部分不足三位, 应补充两个“0”组成三位。小数部分应为 0.100, 用八进制数写成 0.4。所以 205.5 用八进制数表示为 315.4。

把二进制数化成十六进制数, 方法同上。仍然是以小数点为界, 整数部分向左, 小数部分向右, 每四位二进制数为一组, 不足四位的用“0”补充, 再用十六进制数字写出来。

本题中, 整数部分每四位一组为 1100 1101, 写成十六进制数为 CD。小数部分不足四位, 应补足三个“0”组成四位, 小数部分应为 0.1000, 用十六进制数写为 0.8。所以 205.5 用十六进制数表示为 CD.8。

2 用原码、补码、反码表示下列各二进制数(用 8 位表示), 从供选择的答案中选出正确的答案填入 内。

1. -0.1101010

①它的原码表示为 ; ②它的反码表示为 ; ③它的补码表示为 。

2. 0.0010101

①它的原码表示为 ; ②它的反码表示为 ; ③它的补码表示为 .

〔供选择的答案〕

a. 0010101; b. 0110011; c. 11101010; d. 11101011; e. 10010101; f. 11101011; g. 01101010; h. 00010101; i. 10010110.

〔答案〕

1. ①—c; ②—e; ③—i。

2. ①—h; ②—h; ③—h。

〔分析〕

根据原码、补码和反码定义找出正确答案。

①—0.1101010的三种码表示法分别为：

原码：11101010

反码：10010101

补码：10010110

②0.0010101的三种码表示法全为：00010101。

3 已知 $[x]_{\text{补}}$, $[y]_{\text{补}}$, 计算 $[x+y]_{\text{补}}$ 。

从供选择的答案中选用正确的答案填入 内。

1. $[x]_{\text{补}}=0.11011$; $[y]_{\text{补}}=1.01011$;

$[x+y]_{\text{补}}=\text{}$ 。

2. $[x]_{\text{补}}=1.01010$; $[y]_{\text{补}}=1.11111$;

$[x+y]_{\text{补}}=\text{}$ 。

3. $[x]_{\text{补}}=1.01010$; $[y]_{\text{补}}=1.00100$;

$[x+y]_{\text{补}}=\text{}$ 。

〔供选择的答案〕

①10.00110; ②0.01110; ③11.01001;

④0.00110; ⑤10.01110; ⑥1.01001;

⑦溢出。

〔答案〕

1—④; 2—⑥; 3—⑦。

〔分析〕

根据补码加法运算规则可知： $[x]_{\text{补}}+[y]_{\text{补}}=[x+y]_{\text{补}}$ 。

第1小题应为：

$$\begin{array}{r}
 0.11011 \\
 + 1.01011 \\
 \hline
 [1] 0.00110
 \end{array}$$

符号位向高位进位应舍去。

第2小题应为:

$$\begin{array}{r}
 1.01010 \\
 + 1.11111 \\
 \hline
 [1] 1.01001
 \end{array}$$

第3小题应为:

$$\begin{array}{r}
 1.01010 \\
 + 1.00100 \\
 \hline
 [1] 0.01110
 \end{array}$$

结果为溢出。两个相同符号的数相加,结果的符号应与参加运算的任何数符号相同,否则就产生了溢出。

4 已知 $[x]_{\text{反}}$, $[y]_{\text{反}}$, 计算 $[x+y]_{\text{反}}$ 。从供选择的答案中选用正确的答案填入 内。

1. $[x]_{\text{反}}=0.11011$; $[y]_{\text{反}}=1.01010$;

$[x+y]_{\text{反}}=$.

2. $[x]_{\text{反}}=1.01001$; $[y]_{\text{反}}=1.11110$;

$[x+y]_{\text{反}}=$.

3. $[x]_{\text{反}}=1.01001$; $[y]_{\text{反}}=1.00011$;

$[x+y]_{\text{反}}=$.

〔供选择的答案〕

①10.00101; ②1.01000; ③10.01100;

④11.00111; ⑤0.01101; ⑥0.00110;

⑦溢出。

〔答案〕

1 — ⑥; 2 — ②; 3 — ⑦。

〔分析〕

根据 $[x]_{\text{反}}+[y]_{\text{反}}=[x+y]_{\text{反}}$ 这个运算规则进行运算。值得注意的问题是:反码加法运算有一个循环进位的问题。运算结果溢出判断的方法是:当两个操作数的符号相同时,运算结果才有可能溢出,否则没有溢出问题。溢出的判断方法是当两个同号操作数参加运算时,如果结果的符号与操作数的符号相反,则为溢出。

5 用补码一位乘法计算 $[x \cdot y]_{\text{补}}$, 结果只取高五位。从供选答案中选出正确结果填入 中。

1. $x=0.11011$; $y=-0.11111$;

$[x \cdot y]_{\text{补}}=$.

2. $x=-0.01101$; $y=-0.01110$;

$[x \cdot y]_{\text{补}}=$.

〔供选择的答案〕

①11.00101; ②11.11010; ③11.00110;

④00.01011; ⑤00.00101; ⑥11.00101。

〔答案〕

1—①; 2—⑤。

〔分析〕

按照补码一位乘法规则去做, 只取结果的高位部分, 舍去结果的低位部分。

6 用加减交替法计算 $\left[\frac{x}{y}\right]_{\text{补}}$ (不采用恒置“1”的方法), 从供选答案中选出一正确

答案填入 内。

1. $[x]_{\text{补}}=1.01011$; $[y]_{\text{补}}=0.11011$;

$$\left[\frac{x}{y}\right]_{\text{补}} = \text{}。$$

2. $[x]_{\text{补}}=0.10001$; $[y]_{\text{补}}=1.01111$;

$$\left[\frac{x}{y}\right]_{\text{补}} = \text{}。$$

〔可供选择的答案〕

①11.11000; ②00.11000; ③11.00111;

④11.00000; ⑤11.10000; ⑥00.10000。

〔答案〕

1—③; 2—④。

〔分析〕

根据补码一位除法运算规则进行计算, 商只取五位。

7 字长为48位的一个浮点数, 阶码8位用移码表示, 尾数40位 (其中一位符号位) 用补码表示 (基数为2), 从供选择的答案中选出正确答案填入 中。

(1) 最大正数的二进制表示为 , 十进制表示为 。

(2) 最小负数的二进制表示为 , 十进制表示为 。

(3) 最接近于“0”的正数为 。

(4) 它的规格化数所能表示的数的范围为 。

〔可供选择的答案〕

a. $-2^{127} \sim (1-2^{-89}) \times 2^{127}$; b. $-2^{-128} \sim (1-2^{-39}) \times 2^{-127}$;

c. $\underbrace{011\dots 1}_{8\text{个“1”}} \underbrace{11\dots 1}_{39\text{个“1”}}$; d. $\underbrace{111\dots 1}_{48\text{个“1”}}$;

e. $\underbrace{0011\dots 1}_{7\text{个“1”}} \underbrace{11\dots 1}_{39\text{个“1”}}$; f. $\underbrace{010\dots 0}_{7\text{个“0”}} \underbrace{0\dots 0}_{39\text{个“0”}}$;

g. $\underbrace{10\dots 0}_{8\text{个“0”}} \underbrace{0\dots 0}_{39\text{个“0”}}$; h. $\underbrace{00\dots 01}_{47\text{个“0”}}$; i. $\underbrace{0100\dots 0}_{46\text{个“0”}}$ 。

[答案]

(1)—c; ; (2)—g; (3)—h; (4)—a。

[分析]

八位阶码二进制表示以及它所对应的值列于表1-1。

表1-1

阶码 (用移码表示)	阶码所对应的值
11111111	2^{127} (阶码最大)
11111110	2^{126}
⋮	⋮
10000001	2^1
10000000	2^0
01111111	2^{-1}
⋮	⋮
00000001	2^{-127}
00000000	2^{-128} (阶码最小)

40位尾数二进制表示及其所对应的值列于表1-2。

表1-2

符号位	尾数 (用补码表示)	尾数所对应的值
0	$\underbrace{11\cdots\cdots 11}_{39\text{个“1”}}$	$1-2^{-39}$ (最大尾数)
0	$\underbrace{11\cdots\cdots 10}_{38\text{个“1”}}$	$1-2^{-38}$
⋮	⋮	⋮
0	$\underbrace{00\cdots\cdots 01}_{38\text{个“0”}}$	2^{-39} 最接近“0”的正尾数
0	$\underbrace{00\cdots\cdots 0}_{39\text{个“0”}}$	0.0
1	$\underbrace{11\cdots\cdots 1}_{39\text{个“1”}}$	-2^{-39}
⋮	⋮	⋮
1	$\underbrace{00\cdots\cdots 01}_{38\text{个“0”}}$	$-(1-2^{-39})$
1	$\underbrace{00\cdots\cdots 0}_{39\text{个“0”}}$	-1 最小的尾数

规格化的尾数（用补码表示）列于表1-3。

表1-3

符号位	规格化的尾数	所对应的值
0	$\underbrace{11\cdots\cdots 1}_{39\text{个“1”}}$	$1 - 2^{-39}$ 最大尾数
0	$\underbrace{11\cdots\cdots 10}_{38\text{个“1”}}$	$1 - 2^{-38}$
$\vdots$	$\vdots$	$\vdots$
0	$\underbrace{10\cdots\cdots 01}_{37\text{个“0”}}$	$2^{-1} + 2^{-39}$
0	$\underbrace{100\cdots\cdots 0}_{38\text{个“0”}}$	2^{-1}
1	$\underbrace{011\cdots\cdots 1}_{38\text{个“1”}}$	$-(2^{-1} + 2^{-39})$
$\vdots$	$\vdots$	$\vdots$
1	$\underbrace{0\cdots\cdots 01}_{38\text{个“0”}}$	$-(1 - 2^{-39})$
1	$\underbrace{00\cdots\cdots 0}_{39\text{个“0”}}$	-1

通过以上分析，就可以得出正确答案：

最大正数的二进制表示为：0 11..... 1 11..... 1

↑
尾符 8 个“1” 39 个“1”

最大阶码 最大尾数

最小负数的二进制表示为：1 00..... 0 00..... 00

↑
尾符 8 个“0” 39 个“0”

最小阶码 最小尾数

最接近“0”的正数为：0 00..... 0 0..... 01

↑
尾符 8 个“0” 38 个“0”

规格化数的范围： $-1 \times 2^{127} \sim (1 - 2^{-39}) \times 2^{127}$

8 将下列十进制数表示成浮点规格化数（阶码3位，用补码表示。尾数7位，其中一位符号位，用原码表示）。从可选答案中选出正确的答案填入 中。

(1) $-\frac{27}{64}$ 它的浮点规格化数为 的形式。

(2) $-\frac{27}{64}$ 它的浮点规格化数为 的形式。

〔可供选择的答案〕

<i>a</i>	0	101	110110
----------	---	-----	--------

<i>e</i>	1	101	110110
----------	---	-----	--------

<i>b</i>	0	001	011011
----------	---	-----	--------

<i>f</i>	1	111	001010
----------	---	-----	--------

<i>c</i>	0	001	110110
----------	---	-----	--------

<i>g</i>	1	111	110110
----------	---	-----	--------

<i>d</i>	0	111	110110
----------	---	-----	--------

<i>h</i>	1	001	011011
----------	---	-----	--------

↑ ↑ ↑
 尾符 阶码 尾数

〔答案〕

(1)—*d*; (2)—*g*。

〔分析〕

$-\frac{27}{64}$ 的二进制表示为 0.011011, 把它写成规格化数为: 0.11011×2^{-1} 。则它的

浮点数表示格式为:

0	111	110110
---	-----	--------

↑ ↑ ↑
 尾符 阶码 尾数

$-\frac{27}{64}$ 的二进制表示为: -0.011011 , 把它写成规格化数为: -0.11011×2^{-1} 。

则它的浮点数表示格式为:

1	111	110110
---	-----	--------

↑ ↑ ↑
 尾符 阶码 尾数

9 把下列各式化简成最简的“与—或”表达式，从供选择的答案中选出正确答案填入 内。

$$(1) F = \bar{A}\bar{B} + (AB + A\bar{B} + \bar{A}B)C$$

$$= \text{};$$

$$(2) F = \bar{A}B + \bar{A}C + \bar{B}\bar{C} + AD$$

$$= \text{}。$$

〔供选择的答案〕

- a. $\bar{A}\bar{B} + AC + BC$; b. $C + \bar{A}\bar{B}$;
c. $\bar{A} + D + \bar{B}\bar{C}$; d. $\bar{A} + \bar{B}\bar{C} + AD$ 。

〔答案〕

(1)—b; (2)—c。

〔分析〕

$$(1) F = \bar{A}\bar{B} + (AB + A\bar{B} + \bar{A}B)C$$

$$= \bar{A}\bar{B}C + \bar{A}\bar{B}\bar{C} + \underline{\bar{A}BC} + \underline{\bar{A}\bar{B}C} + \underline{\bar{A}BC}$$

$$= \bar{B}C + BC + \bar{A}\bar{B}\bar{C}$$

$$= C + \bar{A}\bar{B}\bar{C}$$

$$= C + \bar{A}\bar{B}$$

$$(2) F = \bar{A}B + \bar{A}C + \bar{B}\bar{C} + AD$$

$$= \bar{A}(B + C) + \bar{B}\bar{C} + AD$$

$$= \bar{A}\bar{B}\bar{C} + \bar{B}\bar{C} + AD$$

$$= \bar{A} + \bar{B}\bar{C} + AD$$

$$= \bar{A} + D + \bar{B}\bar{C}$$

10 某计算机是一台补码运算的机器。试判断下述概念的正确性，并把正确的论述挑出来。

- (1) 冯·诺曼形的计算机是通过数据在内存当中存放的位置来区分指令和数据的。
- (2) 计算机是通过存放在内存存储器中的数据最高位来区分带符号数或不带符号数的。
- (3) 对相同两单元中的数据进行带符号数或不带符号数的比较，都会得到相同的结果。
- (4) 在用机器码填写相对转移指令时，相对位移量一定要以补码形式填入指令字中。
- (5) 在补码运算的机器中，用相同长度的字节数表示带符号数或不带符号数，则所能表示的数字的个数相同。
- (6) 在相同长度的字节数中，带符号数和不带符号数所能表示的数的范围相同。
- (7) 程序计数器和内存地址寄存器是一回事，都指示从内存某个单元取指令或数据。
- (8) 程序计数器指的是从内存单元取指令，内存地址寄存器指的是从内存单元存取数据。

〔答案〕

(5)、(6)。

〔分析〕

一般人们在编写程序时，都是把程序和数在内存中分开存放，但指令和数在内存当中并没有标志区分它们。机器在执行指令时，也不是根据指令和数存放的位置来区分它们，而是根据机器的不同执行周期来决定当前从内存中读出的内容是指令还是数。在取指周期，机器不管从内存的什么地方取来的内容全当作指令送入指令寄存器。而在执行周期，机器从内存中取来的内容一定是操作数。

指令的执行顺序是由程序计数器指出。在取指周期，内存地址寄存器根据程序计数器的值，从内存中把指令取出。而在执行周期，操作数的地址直接给内存地址寄存器，根据内存地址寄存器的值，对操作数进行存取操作。因此，程序计数器只管指令的执行顺序，而内存地址寄存器具体指出对内存的哪个单元进行操作。

至于内存中存放的数，可以看成带符号的数，也可以看成不带符号的数。至于是什么样的数，就看程序员在编程时如何使用它。

例如某两个数 A 和 B ，如果把它们看成带符号的数，则它们的最高位就是它们的符号位。如果把它们看成不带符号的数，那它们的最高位就是它们数值的一部分。当然，相同长度的带符号数和不带符号数表示数的范围不一样。例如八位二进制整数，如用补码表示带符号的数，数的表示范围为 -128 到 $+127$ ，共256个数。如果是不带符号的数，数的表示范围为0到255，共256个数。

带符号数和不带符号数在比较大小时，方法也不同。对于带符号的数，如果是两个同号数比较，利用结果的符号即可判断其大小。例如 A 和 B 两数同号，比较后结果为正，则说明 $A \geq B$ ，反之，则为 $A < B$ 。如果 A 、 B 异号，比较结果就有溢出问题。如果比较结果不溢出，则比较结果正确，判断两数大小与两数同号一样。如果比较结果溢出，比较结果的符号与正确结果的符号相反。如果结果为正，则 $A < B$ ，否则 $A \geq B$ 。

相对位移，一般指目标地址与当前指令所在地址的差。如果目标地址大于当前指令所在地址，差值为正，否则为负。而在做变址运算时，都是当前地址加上相对位移量，得到目标地址，因此相对位移量一定要用补码表示。

11 从供选择的答案中选择应填入 中的正确答案。

下面是关于计算机系统的描述：

(1) 浮点运算指令对用于 A 的计算机是很必要的，可以提高机器的运算速度。

十进制运算指令对有大量输入输出数据的 B 计算机较为有用。

(2) 兼容机之间指令系统 C ；实现方法可以 D 。在同一系列的计算机中，高档机 E 低档机全部指令，其目的使低档机上的程序不加修改即可在高档机上运行。这种性质叫做 F 。

(3) 指令系统由 G 和地址码两部分组成。在一台机器内，各条指令的地址码所包含的地址数叫 H 。

(4) 控制程序流变化的指令叫 I 。

(5) 利用堆栈进行算术逻辑运算的指令可以不设置 J。

〔供选择的答案〕

A, B:

①数据处理 ②工业控制 ③科学计算

C, D, E:

①包括 ②不包括 ③相同 ④不相同

F, G, J:

①向上兼容 ②向下兼容 ③完全兼容

④指令码 ⑤操作码 ⑥地址码

H: ①相同 ②不完全相同

I: ①控制指令 ②转移指令

〔答案〕

A—③ B—① C—③ D—④

E—① F—① G—⑤ H—②

I—② J—⑥

〔分析〕

(3) 算术逻辑运算需要指出 2 或 3 个地址; 转移指令只要一个地址, 停机指令不需要地址。所以不是所有指令的地址码都相同。

(5) 堆栈指令由堆栈指针指出参加运算的存放在存储器相邻单元中的两个数。

12 从下述有关寻址方式的描述中, 从供选择的答案中, 选出应填入 中的正确答案。

假设寄存器字长 16 位, 用十六进制表示。

(1) 在段式管理存储器中, 将段的起始地址存放在基址寄存器中, 今假设基址寄存器的内容为 1000, 变址寄存器的内容为 0400, 指令的一个地址码为 $3F$ 。请求出下列各种情况的有效地址: 若为变址编址, 有效地址为 A; 若为直接编址, 有效地址为 B; 若为相对编址, 地址为 C (本条指令的实际地址为 2000)。

(2) 假如不是段式管理, 不设基址寄存器, 其余情况同 (1), 如为变址编址, 则有效地址为 D。

(3) 如指令地址码部分为参与运算的数, 称为 E。

〔供选择的答案〕

A, B, C, D:

① 1000; ② $103F$; ③ $143F$; ④ $043F$; ⑤ $003F$;

⑥ $243F$; ⑦ $203F$ 。

E: ①运算数; ②立即数。

〔答案〕

A—③; B—②; C—⑦; D—④; E—②。

〔分析〕