

一、选择题

1. 下列各种数制的数中, 最小的数是(B)。

- A. (101001)B B. (101001)BCD
C. (52)0 D. (233)H

A 为 29H, B 为 29D. C 写成二进制为 101010, 即 2AH, 显然最小的为 29D。

2. 两个数 7E5H 和 4D3H 相加, 得(C)。

- A. BD8H B. CD8H
C. CB8H D. CC8H

在十六进制数的加减法中, 逢十六进一, 故而 $7E5H + 4D3H = CB8H$ 。

3. 一个 16 位无符号二进制数的表示范围是(B)。

- A. 0—5536 B. 0~66535 C. -32768—32767 D. -32768~32768

一个 16 位无符号二进制数的表示范围是 0. -- 216_1。

4. 下列校验码中, 奇校验正确的有(C)。

- A. 110100111 B. 001000111
C. 010110011 D. 110100111

选项 A、B、D 中“1”的个数为偶数, 仅有选项 C 中“1”的个数为奇数。

5. 能发现两位错误并能纠正 1 位错的编码是(B)。

- A. CRC 码 B. 海明码
C. 偶校验码 D. 奇校验码

偶校验码与奇校验码都不能纠错; CRC 码可以发现并纠正信息串行读 / 写、存储或传送中出现的 1 位或多位错 (与多项式的选取有关): 海明码能发现两位错误并纠正 1 位错。

6. 设待校验的数据为 $D_8 \sim D_1 = 10101011$, 若采用海明校验, 其海明码为(A)(设海明码具有一位纠错能力, P_{13} 采用全校验); 若采用 CRC. 且生成多项式为 10011, 则: CRC 码为(C)。

- A. 0 1010 0 101 1 1 11 B. 0 1000 0 111 1 1 11
C. 10101011 1010 D. 10101010 1011

当采用海明校验时, 海明码为 $P_{13} - PI: Q \ 1010 \ 0101 \ 1111$ (下画线的为校验位)。其中

$PI = P_{38}P_5(DP_7 @ P_9 \sim P_{11} - I, P_2 = P_{30}P_{60}P_7QPIOP_{11} = 1,$

$P_4 = P_5QP_{60}P_7 \sim P_{12} - 1, P_8 - P_{90}PIOP_{11}OP_{12} = 0$ 。 P_{13} 位为全校验位, 因为 $P_{12} - PI$ 中 1 的个数为偶数个, 故 $PI_3 \sim D$: 采用 CRC 时-将信息位左移 4 位 t 进行模 2 除, 得余数为 1010, 故 CRC 码为 10101011 1010。

7. 一个 $n+1$ 位整数 x 原码的数值范围是(D)。

- A. $-2^{n+1} < x < 2^{n+1} - 1$ B. $-2^n + 1 \leq x < 2^n - 1$ C. $-2^n + 1 < x \leq 2^n - 1$ D. $-2^n + 1 \leq x \leq 2^n - 1$
 $n+1$ 位整数原码的表示范围为 $-2^{n+1} \leq x \leq 2^{n+1} - 1$ 。

8. 补码定点整数 01010101 左移两位后的值为(B)。

- A. 0100 0111 B. 0101 0100 C. 0100 0110 D. 0101 0101

该数是一个正数, 按照算术补码移位规则, 正数左右移使均添 0, 且符号位不变, 所以 0101 0101 左移 2 位后的值为 0101 0100。

9. 补码定点整数 1001 0101 右移 1 位后的值为(D)。

- A. 0100 1010 B. 010010101 C. 1000 1010 D. 1100 1010

该数是一个负数，按照算术补码移位规则，负数右移添 1，负数左移添 0。所以 1001 0101 右移 1 位后的值为 1100 1010。

10. 原码乘法是(A)。

- A. 先取操作数绝对值相乘，符号位单独处理
- B. 用原码表示操作数，然后壹接相乘
- C. 被乘数用原码表示，乘数去绝对值，然后相乘
- D. 乘数用原码表示，被乘数去绝对值，然后相乘

原码一位乘法中，符号位与数值位是分开进行运算的。运算结果的数值部分是乘数与被乘数数值位的乘积，符号是乘数与被乘数符号位的异或。

11. 在原码一位乘法中。(B)。

- A. 符号位参加运算
- B. 符号位不参加运算
- C. 符号位参加运算，并根据运算结果改变结果中的符号位
- D. 符号位不参加运算，并根据运算结果确定结果中的符号

在原码一位乘法中，符号位不参加运算，符号位单独处理，同号为正，异号为负

12. 实现 N 位（不包括符号位）补码一位乘时，乘积为(C)位

- A. ,N
- B. N+1
- C. 2N
- D. 2N+1

补码一位乘法运算过程中一共向右移位 N 次，加上原先的 N 位，一共是 2N 位数值位。

13. 下列关于补码除法说法正确的是(B)。

- A. 补码不恢复除法中，够减商 0，不够减商 1
- B. 补码不恢复余数除法中，异号相除时，够减商 0j 不够减商 1
- C. 补码不恢复除法中，够减商 1，不够减商 0
- D. 以上都不对

补码除法（不恢复余数法 / 加减交替法）。异号相除是看够不够减，然后上商，够减则商 0。不够减商 1

14. 设[X]补=0.1011 【Y】补=1.1110，求【X+Y】补和【X-Y】补的值。

【X+Y】补=0.1011+ 1.1110=0.1001, [X-Y] 补=0.1011+0.0010=0.1101

15. 某浮点机，采用规格化浮点数表示，阶码用移码表示（最高位代表符号位）。尾数用原码表示。下列哪个数的表示不是规格化浮点数？(B)

阶码 尾数 阶码 尾数

原码表示时，首位为 1 是规格化浮点数。

- A. 11111111,1.1000...00
- B. 00111111,1.0111...01
- C. 1000001,0.1111...01
- D. 01111111,0.1000...10

16. 浮点数的 IEEE 754 标准对尾数编码采用的是(A)。

- A. 原码
- B. 反码
- C. 补码
- D. 移码

IEEE 754 标准中尾数采用原码表示，且短浮点数、长浮点数采用隐藏位，临时浮点数不采用隐藏位方案。

17. 在浮点数编码表示中，(D)在机器数中不出现，是隐含的。

- A. 阶码
- B. 符号
- C. 尾数
- D. 基数

浮点数表示中基数的值是约定好的，故隐含。

18. 已知两个实数 $x=-68$, $y=-8.25$. 它们在 C 语言中定义为 float 型变量，分别存放在寄存器 A 和 B 中，另外，还有两个寄存器 C 和 D。A、B、C、D 都是 32 位的寄存器。请问（要求用十六进制表示二进制序列）：

1) 寄存器 A 和 B 中的内容分别是什么？

2) x 和 y 相加后的结果存放在 C 寄存器中，寄存器 C 中的内容是什么？

3) x 和 y 相减后的结果存放在 D 寄存器中，寄存器 D 中的内容是什么？

解答：

1) float 型变量在计算机中都被表示成 IEEE 754 单精度格式。 $X=-68=-(1000100)_2=-1.0001 \times 2^6$ ，符号位为 1，阶码为 $127+6=128+5=(1000\ 0101)_2$ ，尾数为 1.0001。所以小数部分为：000 1000 0000 0000 0000 0000。合起来整个浮点数表示为：1 1000 0101 000 1000 0000 00000000 0000。写成十六进制为：C2880000H。

$Y=-8.25=-(100.01)_2=-1.00001 \times 2^3$ ，符号位为 1，阶码为 $127+3=128+2=(1000\ 0010)_2$ ，尾数为 1.00001，所以小数部分为：000 0100 0000 0000 0000 0000，合起来整个浮点数表示为：1 1000 0010 000 0100 0000 0000 0000 0000 写成十六进制为 C1040000H。

因此，寄存器 A 和 B 的内容分别为 C2880000H、C1040000H。

2) 两个浮点数相加的步骤如下。

① 对阶： $E_x=10000101$, $E_y=10000010$ ，则：

$【E_x - E_y】补 = 【E_x】补 + 【-E_y】补 = 10000101 + 0111\ 1110 = 00000011$ 。

E_x 大于 E_y ，所以对 y 进行对阶。对阶后， $Y=-0.00100001 \times 2^6$ 。

② 尾数相加： x 的尾数为 -1.000 1 000 0000 0000 0000 0000， y 的尾数为 -0.001 0000 10000000 0000 0000，

用原码加法运算实现，两数符号相同，做加法，结果为 -1.001 1000 1000 0000 0000 0000。

即 x 加 y 的结果为 -1.001 1000 $\times 2^6$ ，所以符号位为 1，尾数为：001 1000 1000 0000 00000000。阶码为 $127+6=128+5$ ，即：1000 0101。合起来为：11000 0101 001 1000 1000 0000 00000000。转换为十六进制形式为：C2988000H。

所以 C 寄存器中的内容是 C2988000H。

3) 两个浮点数相减的步骤同加法，对阶的结果也一样，只是尾数相减。

尾数相减： x 的尾数为 -1.000 1000 0000 0000 0000 0000， y 的尾数为 -0.001 0000 1000 000000000000。

用原码减法运算实现，两数符号相同，做减法：符号位：取大数的符号，负数，为 1；数值部分：大数减小数负数的补码：

1.	000	1000	0000	0000	0000	0000
+	1.	110	1111	1000	0000	0000
	0.	111	0111	1000	0000	0000

所以：

符号位为 1，尾数为 1101 111 0000 0000 Cb000 0000。阶码为 $127+5=128+4$ ，即 10000100。合起来为：1 1000 0100 110 1111 0000 0000 0000 0000。转换为十六进制形式为：C26FOOOOH。所以寄存器 D 中的内容是 C26FOOOOH。

19. 加法器采用并行进位的目的是(C)。

A. 增强加法器功能 B. 简化加法器设计

C. 提高加法器运算速度 D. 保证加法器可靠性
并行进位可以大大提高加法器的运算速度。