

# 第一章 计算机组成原理

## 1.1 选择题

- 冯·诺依曼机的基本工作方式是()  
A. 控制流驱动方式      B. 多指令多数据流方式  
C. 微程序控制器      C. 数据流驱动方式
- ▲ 将高级语言源程序转换为机器级目标文件的程序是()  
A. 汇编程序      B. 连接程序      C. 编译程序      D. 解释程序
- 在计算机中,CPU 的 CPI 与下列()因素无关.  
A. 时钟频率      B. 系统结构      C. 指令集      D. 计算机组织
- 某计算机主频为  $1GHz$ , 程序 P 运行过程中, 共执行了 10000 条指令, 其中,80% 的指令执行平均需要 1 个时钟周期,20% 的指令执行平均需 10 个时钟周期. 程序 P 的平均 CPI 和 CPU 执行时间分别是()  
A. 2.8,  $28\mu s$       B. 28,  $28\mu s$       C. 2.8,  $28ms$       D. 28,  $28ms$
- 若  $X$  为负数, 则由  $[X]_{\text{补}}$  求  $[-X]_{\text{补}}$  是将()  
A.  $[X]_{\text{补}}$  各值保持不变  
B.  $[X]_{\text{补}}$  符号位变反, 其他位不变  
C.  $[X]_{\text{补}}$  除符号位外, 其余位取反, 末尾加一  
D.  $[X]_{\text{补}}$  连同符号位一起变反, 末尾加一
- 对于相同位数 (设  $N$  位, 不考虑符号位) 的二进制补码小数和十进制小数, 二进制小数能表示的数的个数/十进制小数所能表示的个数为()  
A.  $(0.2)^N$       B.  $(0.2)^{N-1}$       C.  $(0.02)^N$       D.  $(0.02)^{N-1}$
- 设  $x$  为真值,  $x^*$  为其绝对值, 满足  $[-x^*]_{\text{补}} = [-x]_{\text{补}}$  当且仅当  $x$  为()

- A. 任意数      B. 正数      C. 负数      D. 以上说法均不正确
8. ALU 作为运算器的核心部件, 其属于 ( )
- A. 时序逻辑电路      B. 组合逻辑电路      C. 控制器      D. 寄存器
9. ◆ 加法器中每位的进位信号由 ( ) 组成
- A.  $X_i \oplus Y_i$       B.  $X_i Y_i$       C.  $X_i Y_i C_i$       D.  $X_i + Y_i + C_i$
10. 一个 8 位寄存器内的数值为 1100 1010 进位标志位寄存器 C 为 0, 若将此 8 位寄存器循环左移 1 位后, 则该寄存器和标志寄存器的值分别为 ( )
- A. 1001 0100, 1      B. 1001 0101, 0      C. 1001 0101, 1      D. 1001 0100, 0
11. 设机器数字长 8 位 (含一位符号位), 若机器数为 *BAH* 为原码, 算术左移 1 和算术右移 1 位分别得 ( )
- A. *F4H, EDH*      B. *B4H, 6DH*      C. *74H, DDH*      D. *B5H, EDH*
12. 关于模 4 补码, 下列说法中正确的是 ( )
- A. 模 4 补码和模 2 补码不同, 它不容易检查乘除运算中的溢出问题
- B. 每个模 4 补码存储时只需要存储一位符号位
- C. 存储每个模 4 补码需要存储两个符号位
- D. 模 4 补码, 在算术与逻辑部件中为一个符号位
13. 在原码一位乘法中, ( )
- A. 符号位参与运算
- B. 符号位不参与运算
- C. 符号位参与运算, 并根据运算结果改变结果中的符号位
- D. 符号位不参与运算, 并根据运算结果改变结果中的符号位
14. ◆ 某计算机字长为 8 位, CPU 中有一个 8 位加法器. 已知无符号数  $x = 69, y = 38$ , 如果在该加法器中计算  $x - y$ , 则加法器的两个输入端入端信息和低位进位信息分别是 ( )
- A. 0100 0101, 0010 0110, 0      B. 0100 0101, 1101 1001, 1
- C. 0100 0101, 1101 1010, 0      D. 0100 0101, 1101 1010, 1
15. ▲ 某计算机存储器按字节编制, 采用小端方式存放数据. 假定编译器规定 *int* 型和 *short* 型长度分别为 32 位和 16 位并且数据按边界对齐存储. 某 C 语言程序段如下

```
struct {
```

```
int a;  
char b;  
short c;  
}  
record;  
record.a = 273;
```

若 record 变量的首地址为 0xC008 地址 0xC008 中的内容及 record.c 的地址分别是 ()

A. 0x00, 0xC00D      B. 0x00, 0xC00E      C. 0x11, 0xC00D      D. 0x11, 0xC00E

16. ▲ 有如下 C 语言序段:

```
short si = -32767;  
unsigned short usi = si;
```

这执行上述两条语句后,usi 的值是 \_\_\_\_

17. 某计算机字长为 32 位,按字节编址,采用小端方式存放数据,假定有一个 double 型变量,其机器数表示为 1122 3344 5566 7788H,存放在以 0000 8040H 开始的连续存储单元中,则存储单元 0000 8046H 中存储的是 ()

A. 22H      B. 33H      C. 77H      D. 66H

18. 在规格化浮点运算中,若浮点数  $2^5 \times 1.10101$ ,其中尾数为补码表示,则该数 ()

A. 不需要规格化      B. 需要右移规格化  
C. 需将尾数左移一位规格化      D. 需将尾数左移二位规格化

19. 某浮点机,采用规格化浮点数表示,阶码用移码表示 (最高位表示符号位),尾数用原码表示,下列 () 表示不是规格化浮点数

A. 1111111, 1.10000...000      B. 0011111, 1.0111...01  
B. 1000001, 0.11111...111      D. 0111111, 0.100000000

20. 下列关于对阶操作说法正确的是 ()

A. 在浮点数加减运算对阶操作中,若阶码减少,则尾数左移  
B. 在浮点数加减运算对阶操作中,若阶码增大,则尾数右移;若阶码减少,则尾数左移  
C. 在浮点数加减运算对阶操作中,若阶码增大,则尾数右移  
D. 以上说法都不对

21. 在 IEEE 754 标准中,它所能表示的最小规格化负数为 ()

22. 采用规格化的浮点数最主要是为了 ( )
- A. 增加数据的表示范围      B. 方便浮点运算  
C. 防止运算时数据溢出      D. 增加数据的表示精度
23. 设浮点数共 12 位, 其中阶码以 4 位补码表示 (1 位符号), 尾数用 8 位补码表示 (1 为符号). 则该规格化浮点数所能表示的最大正数为 ( )
24. 若浮点数的尾数用补码表示, 则下列 ( ) 中的尾数是规格化形式
- A. 1.11000      B. 0.01110      C. 0.01010      D. 1.00010
25. 设浮点数的基数为 4, 尾数用原码表示, 则以下 ( ) 是规格化的数
- A. 1.001101      B. 0.001101      C. 1.011011      D. 0.000010
26. 下列关于舍入的说法, 正确的是 (多选)( )
- (1) 不仅仅只有浮点数需要舍入, 定点数在运算时也可能舍入  
(2) 在浮点数舍入中, 只有左规格化时可能舍入  
(3) 在浮点数舍入中, 只有右规格化时可能舍入  
(4) 在浮点数舍入中, 左, 右规格化时都可能舍入  
(5) 舍入不一定能产生误差
27. ▲ 假定变量  $i, f, d$  的数据类型分别是  $int, float, double$  ( $int$  用补码表示, 其余用 IEEE 754 浮点数格式), 已知  $i = 785, f = 1.5678E3, d = 1.5E100$ , 若在 32 位机器中执行下列关系表达式, 则结果为真的是 (多选)( )
- (1)  $i == (int)(float)i$   
(2)  $f == (float)(int)f$   
(3)  $f == (float)(double)f$   
(4)  $(d + f) - d == f$
28. ▲  $float$  类型数据通常用 IEEE 754 单精度格式表示, 若编译器将  $float$  型变量  $x$  分配在一个 32 位浮点寄存器  $FR1$  中, 且  $x = -8.25$  则  $FR1$  中的内容是 \_\_\_\_\_
29. ▲ 下列关于浮点数加减运算中, 正确的是 (多选)( )
- (1) 对阶操作不会引起阶码上溢或下溢  
(2) 右规和尾数舍入都可以引起阶码上溢

- (3) 左规时可能引起阶码下溢
- (4) 尾数溢出时结果不一定溢出
30. ▲ 假定用若干  $2K \times 4$  的芯片组成一个  $8K \times 8$  的存储器, 则地址 081FH 所在芯片的最小地址是 ( )
- A.0000H    B.0600H    C.0700H    D.0800H
31. ▲ 某计算机存储器按字节编址, 主存地址空间大小为 64MB, 现用  $4M \times 8$  位的 RAM 芯片组成 32MB 的主存储器, 则存储器地址寄存器 MAR 的位数至少是 \_\_\_\_\_
32. ▲ 某磁盘的转速为 10000 转/分, 平均寻道时间是 6ms, 磁盘传输速率是 20MB/s 磁盘控制器延迟为 0.2ms, 读取一个 4KB 的扇区所需要的平均时间约为 ( )
- A.9ms    B.9.4ms    C.12ms    D.12.4ms
33. ▲ 假设主存地址为 32 位, 按字节编址, 主存和 Cache 之间采用直接映射方式, 主存块大小为 4 个字, 每个字 32 位, 采用回写方式, 则能存放 4K 字数据的 Cache 总容量的位数至少是 ( )
- A.146K    B.147K    C.148K    D.158K
34. ◆ 一个计算机系统采用 32 位单字长指令, 地址码 12 位, 若定义了 250 条二地址指令, 则还可以有 ( ) 单地址指令.
- A.4K    B.8K    C.16K    D.24K
35. ▲ 下列选项中, 属于指令集体系结构 (ISA) 规定的内容是 (多选)( )
- (1) 指令字格式和指令类型
- (2) CPU 的时钟周期
- (3) 同样寄存器个数和位数
- (4) 加法器的进位方式
36. ▲ 设计某指令系统时, 假设采用 16 位定长指令格式, 操作码使用拓展编码方式, 地址码为 6 位, 包括零地址, 一地址和二地址三种指令. 若二地址指令有 12 条, 一地址指令有 254 条, 则零地址指令的条数最多为 ( )
- A.0    B.2    C.64    D.128
37. 指令系统中采用不同寻址方式的目的是 ( )
- A. 提供拓展操作码的可能性并降低译码难度

- B. 可缩短指令字长, 托大寻址空间, 提高编程的灵活性  
C. 实现程序控制  
D. 三者都正确
38. 简化地址结构的基本方法是尽量采用 ( )  
A. 寄存器寻址    B. 隐含寻址    C. 直接寻址    D. 间接寻址
39. 在多道程序设计中, 最重要的寻址方式是 ( )  
A. 相对寻址    B. 间接寻址    C. 立即寻址    D. 按内容寻址
40. 设相对寻址的转移指令占 3B, 第一字节为操作码, 第二、三字节为相对位移量 (补码表示), 而且数据在存储器中采用以低字节为字地址的存放方式。每当 CPU 从存储器取出一字节时, 即自动完成  $(PC)+1 \rightarrow PC$ 。若 PC 的当前值为 240 (十进制), 要求转移到 290 (十进制), 则转移指令的第二、三字节的机器代码是 (); 若 PC 的当前值为 240 (十进制), 要求转移到 200 (十进制), 则转移指令的第二、三字节的机器代码是 ()。  
A. 2FH, FFH    B. D5H, 00H    C. D5H, FFH    D. 2FH, 00H
41. 某计算机有 16 个通用寄存器, 采用 32 位定长指令字, 操作码字段 (含寻址方式位) 为 8 位, Store 指令的源操作数和目的操作数分别采用寄存器直接寻址和基址寻址方式。若基址寄存器可使用任意一个通用寄存器, 且偏移量用补码表示, 则 Store 指令中偏移量的取值范围是 ( )  
A.  $-32768 \sim +32767$     B.  $-32767 \sim +32768$   
C.  $-65536 \sim +65535$     D.  $-65535 \sim +65536$
42. 按字节编址的计算机中, 某 double 型数组 A 的首地址为 2000H, 使用变址寻址和循环结构访问数组 A, 保存数组下标的变址寄存器的初值为 0, 每次循环取一个数组元素, 其偏移地址为变址值乘以 `sizeof(double)`, 取完后变址寄存器的内容自动加 1。若某次循环所取元素的地址为 2100H, 则进入该次循环时变址寄存器的内容是 ( )  
A. 25    B. 32    C. 64    D. 100

## 1.2 综合题