

# 2025年4月 真题解析

# 13015 计算机系统原理

助力2510考期

讲师：小飞学长Pro

**绝密 ★ 考试结束前**

**2025 年 4 月高等教育自学考试**

**计算机系统原理试题**

**课程代码:13015**

# 一、单选题

# 一、单选题（每小题1分）

1. 下列数中最小的数是

A.  $(101001)_2$

B.  $(52)_8$

C.  $(1010001)_{BCD}$

D.  $(233)_{16}$

答案: A

解析: 按“权”展开

A:  $(101001)_2 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 32 + 8 + 1 = 41$

B:  $(52)_8 = 5 \times 8^1 + 2 \times 8^0 = 40 + 2 = 42$

C:  $(1010001)_{BCD}$  是 BCD (二进制编码的十进制) 表示。BCD 每4位代表一个十进制数字，不足位时在左侧补零：1010001 补零为 01010001，分组为 0101 (5) 和 0001 (1)，因此值为 51。

D:  $(233)_{16} = 2 \times 16^2 + 3 \times 16^1 + 3 \times 16^0 = 512 + 48 + 3 = 563$

# 一、单选题（每小题1分）

2. 计算机系统中采用补码运算的目的是
- A. 与手工运算方式保持一致
  - B. 提高运算速度
  - C. 提高运算的精度
  - D. 简化计算机的设计

答案: D

解析:

补码允许加法和减法使用相同的硬件电路（减法变加法，无需单独减法电路）

# 一、单选题（每小题1分）

3. 下列说法不正确的是

- A. 变址寻址时,有效数据存放在内存中
- B. 数据交换指令,将两个寄存器内容互换
- C. 堆栈指针 SP 的内容表示当前堆栈内所存储的数据的个数
- D. 内存中指令的寻址和数据的寻址是交替进行的

答案: C

解析:

SP 是栈顶指针, 指向栈顶的地址, 不是数据个数

# 一、单选题（每小题1分）

4. 通用寄存器用于临时存放从主存取来的数据或运算的结果，下列不属于通用寄存器的部件是
- A. 指令寄存器
  - B. 标志寄存器
  - C. 程序计数器
  - D. 状态计数器

答案：D

解析：【来源：教材P27】

为了临时存放从主存取来的数据或运算的结果，还需要若干**通用寄存器**（General Purpose Register）**组成通用寄存器组（GPRs）**

标志信息需要记录在专门的**标志寄存器**中；

从主存取来的指令需要临时保存在**指令寄存器**（Instruction Register, IR）中；

CPU 为了自动按序读取主存中的指令，还需要有一个**程序计数器**（Program Counter, PC）

**没有状态计数器**的概念

# 一、单选题（每小题1分）

5. 下列不属于指令集体系结构设计所追求的目标的是

- A. 提高机器级程序的执行速度
- B. 增大控制存储器的容量
- C. 缩短机器级指令的长度
- D. 提高机器级程序设计的灵活性

答案：B

解析：

指令集设计目标：提高速度、缩短指令长度、灵活，**不包括**增大控制存储器容量

# 一、单选题（每小题1分）

6. 程序使用链接的好处不包括

- A. 模块化
- B. 结构化
- C. 效率高
- D. 空间利用率高

答案：B

解析：【来源：教材P162】

使用链接的好处

- 1) 模块化
- 2) 效率高
- 3) 提高空间利用率

# 一、单选题（每小题1分）

7. 属于内部异常的是

- A. 采样计时时间到
- B. 网络数据包到达
- C. 非法操作码
- D. 用户按下(Ctrl + C)键

答案: C

解析:

内部异常是由 CPU 内部事件引起，如：非法操作码

其他是外部中断

# 一、单选题（每小题1分）

8. 在存储器层次化体系结构中,以下存取时间最小的存储器是

- A. 寄存器
- B. 硬盘
- C. 主存储器
- D. 高速缓冲存储器

答案: A

解析:

存储器层次从存取速度由快到慢为: 寄存器 → 高速缓冲存储器 (Cache) → 主存储器 (内存) → 硬盘 (外存)。

# 一、单选题（每小题1分）

9. 下列不是 I/O 子系统特性的是

- A. 共享性
- B. 同步性
- C. 复杂性
- D. 异步性

答案：B

解析：【来源：教材P249】

I/O 子系统特性：共享性、复杂性、异步性，无同步性。

# 一、单选题（每小题1分）

10. 现代操作系统主要采用的虚拟存储管理方式是

- A. 页式
- B. 段式
- C. 段页式
- D. 随机式

答案: A

解析: 【来源: 教材P229】

现代操作系统主要采用**页式**虚拟存储管理方式

## 二、填空題

## 二、填空题（每小题2分）

11. 从抽象层次上来分,程序设计语言可分为\_\_\_\_\_语言和\_\_\_\_\_语言。

答案:

高级、低级（答案顺序可对调）

## 二、填空题（每小题2分）

12. CPU 访问主存时,需先将主存地址、读/写命令分别送到总线的\_\_\_\_\_、控制线,然后通过总线的\_\_\_\_\_发送或接收数据。

答案:

地址线、数据线

解析: 【来源: 教材P27】

CPU 访问主存时, 需先将主存地址、读/写命令分别送到总线的地址线、控制线, 然后通过数据线发送或接收数据。

总结: 总线分为**地址线** (传输主存地址) 、**控制线** (传输读/写等控制命令) 、**数据线** (传输数据)

## 二、填空题（每小题2分）

13. 浮点数加减运算过程中,需要经过对阶、\_\_\_\_\_加减、\_\_\_\_\_和舍入4个步骤。

答案：

尾数、规格化

## 二、填空题（每小题2分）

14. 冯·诺依曼结构计算机由运算器、\_\_\_\_\_、\_\_\_\_\_、输入设备和输出设备五大基本部件组成。

答案：

控制器、存储器（答案顺序可对调）

## 二、填空题（每小题2分）

15. 栈是一种采用\_\_\_\_\_方式进行访问的一块存储区，在执行 pushw % ax 指令之后，SP 指向存放有 AX 内容的单元，即当前刚入栈的\_\_\_\_\_。

答案：

前进后出、数据

解析：【来源：教材P107】

栈（Stack）是一种采用“先进后出”方式进行访问的一块存储区，在处理过程调用时非常有用。大多数情况下，栈是从高地址向低地址增长的，在 IA - 32 中，用 ESP 寄存器指向当前栈顶，而栈底通常在一个固定的高地址上。在执行 pushw % ax 指令之后，SP 指向存放有 AX 内容的单元，也即新栈顶指向了当前刚入栈的数据。若随后再执行 popw % ax 指令，则原先在栈顶的两个字节退出栈，栈顶向高地址移动两个单元，又回到 pushw % ax 指令执行前的位置。

## 二、填空题（每小题2分）

16. 将高级语言源程序转换为可执行文件通常分为\_\_\_\_\_、编译、汇编和\_\_\_\_\_4步。

答案：

预处理、链接

## 二、填空题（每小题2分）

17. 动态链接有两种方式,一种是在程序\_\_\_\_\_过程中加载和链接共享库,另一种是在程序\_\_\_\_\_过程中加载并链接共享库。

答案:

**加载、执行 (答案顺序可对调)**

解析: 【来源: 教材P181】

动态链接有两种方式, 一种是在程序**加载**过程中加载和链接共享库, 另一种是在程序**执行**过程中加载并链接共享库。

## 二、填空题（每小题2分）

18. 程序访问的局部性包括\_\_\_\_\_局部性和\_\_\_\_\_局部性。

答案：

时间、空间（答案顺序可对调）

## 二、填空题（每小题2分）

19. CPU 访存过程中,若出现 cache 缺失,由 \_\_\_\_\_ 处理;若出现缺页,由 \_\_\_\_\_ 处理。

答案:

硬件、软件

解析: 【来源: 教材P235】

cache 缺失由硬件处理; 缺页由软件处理, 操作系统通过缺页异常处理程序来实现

## 二、填空题（每小题2分）

20. 因为 I/O 统一编址方式下 I/O 访问和主存访问共用同一组\_\_\_\_\_，所以其保护机制可由\_\_\_\_\_管理机制实现。

答案：

指令、虚拟存储

解析：【来源：教材P281】

因为统一编址方式下I/O访问和主存访问共用同一组指令，所以其保护机制可由虚拟存储管理机制实现。

### 三、名词解释题

### 三、名词解释题（每小题3分）

21. 时钟周期

22. 数据通路

答案：

**21. 时钟周期：**计算机执行一条指令的过程被分成若干步骤完成，每一步都要有相应的控制信号进行控制，计算机必须能够产生同步的时钟定时信号，即 CPU 主脉冲信号，其宽度称为时钟周期。

**22. 数据通路：**通常将指令执行过程中数据所经过的路径，包括路径上的部件称为数据通路。

## 四、简答题

## 四、简答题（每小题6分）

23. 简述字和字长概念。
24. 简述栈在处理过程调用时的作用，并列举 16 位架构下可用的栈相关指令。
25. 简述 CPU 执行一条指令的大致过程。
26. 简述文件系统所实现的功能。

答案：

23. **字**用来表示被处理信息的单位，用于度量各种数据类型的宽度。**字长**是指 CPU 内部用于整数运算的数据通路的宽度。
24. 栈是一种采用“先进后出”方式进行访问的一块存储区，在处理过程调用时非常有用。大多数情况下，栈从高地址向低地址增长。16 位架构下有 pushw 和 popw 指令分别表示进栈和出栈。

## 四、简答题（每小题6分）

23. 简述字和字长概念。
24. 简述栈在处理过程调用时的作用，并列举 16 位架构下可用的栈相关指令。
25. 简述 CPU 执行一条指令的大致过程。
26. 简述文件系统所实现的功能。

答案：

25. 取指令、指令译码、计算源操作数地址并取操作数、执行数据操作、计算目的操作数地址并存结果、计算下条指令地址。
26. 文件系统要为上层的用户和应用程序提供文件抽象以及文件的创建、打开、读 / 写和关闭等所有操作接口；另一方面，文件系统需要将抽象的文件标识与具体的硬件设备建立关联，并通过相应的设备驱动程序实现系统调用接口规定的操作。

## 五、计算题

## 五、计算题（每小题10分）

27. 假设某机器 M 的时钟频率为 4GHz, 用户程序 P 在 M 上的指令条数为  $8 \times 10^8$ , 其 CPI 为 1.25, 则 P 在 M 上的执行时间是多少(请写出计算公式)? 若在机器 M 上从程序 P 开始启动到执行结束所需的时间是 4s, 则 P 的用户 CPU 时间所占的百分比是多少?

**温馨提示:** 第一章课后习题P45第9题(**原题**)。

**答案:**

**用户 CPU 时间 = 程序总时钟周期数 ÷ 时钟频率**

**程序总时钟周期数 = 程序总指令条数 × CPI =  $8 \times 10^8 \times 1.25$**

**所以: 用户 CPU 时间 =  $8 \times 10^8 \times 1.25 \div (4 \times 10^9) = 0.25s$ 。**

**若在机器 M 上从程序 P 开始启动到执行结束所需的时间是 4s,**

**则 P 的用户 CPU 时间所占的百分比是  $0.25 \div 4 = 6.25\%$ 。**

## 五、计算题（每小题10分）

28. (1)若前端总线(FSB)的时钟频率为333MHz,采用4倍并发技术传输数据,总线数据宽度为64位,则工作频率和总线带宽各为多少? (2)若QPI总线时钟频率为2.4GHz,每个时钟周期传输2次数据,其有效数据位16位,则其速度为多少GT/s? 工作频率是多少? 总带宽是多少?

**温馨提示:** FSB + QPI知识 P277 / 第六章课后习题P289第7、8题(原题-每个题各一半题)。

**答案:**

(1) **一般习惯说:** 前端总线(FSB)的工作频率为1333MHz (实际时钟频率为333MHz)

工作频率: **时钟频率×倍数**= $333\text{MHz} \times 4 = 1332\text{MHz}$  (或1333MHz)

总线带宽: **工作频率×总线宽度**= $1332\text{MHz} \times 64\text{位}/8\text{位} = 10.656\text{GB/s}$  (或10.664GB/s)

(2) **公式:** QPI总线的带宽 = 每秒传输次数 × 每次传输的有效数据[B] × 2

速度:  $2.4\text{GHz} \times 2 = 4.8\text{GT/s}$ , 工作频率为 4.8GT/s, [4.8GT/s]表示**每秒钟传输4.8G次数据**

总线带宽:  $4.8\text{GT/s} \times 16\text{位}/8\text{位} \times 2 = 19.2\text{GB/s}$

## 六、分析设计题

## 六、分析设计题（每小题10分）

29. 假设 R[eax] = 0000 00A0H, R[ecx] = 0000 0003H, 请问：

执行指令“mulb % cl”后，哪些寄存器的内容会发生变化？与执行“imulb % cl”指令所发生的变化是否一样？为什么？两条指令得到的 CF 和 OF 标志各是什么？请用该例给出的数据验证你的结论。

**温馨提示：第三章 例3.5 P112(原题第①小问，只是换了R的值)。**

**答案：**

因为 R[eax] = 0000 00**A0H**, R[ecx] = 0000 00**03H**, 所以, R[al] = **A0H**, [cl] = **03H**。

指令 "mulb % cl" 中指出的操作数为 8 位，故指令的功能为 "**R[ax]←R[al]×R[cl]**"，

因此，改变内容的**寄存器是 AX**，指令执行后 R[ax] = **01E0H**，即十进制数 480，

因为高 8 位乘积不为全 0，故 CF 和 OF 标志全为 1。

**扩展：**若结果的高 8 位 (ah) **不全为 0**，说明低 8 位 (al) **存不下**完整结果，需要用高 8 位扩展，此时 CF 和 OF 会被置为 1；若高 8 位**全为 0**，说明低 8 位**能存下**结果，CF 和 OF 置为 0。

## 六、分析设计题（每小题10分）

答案：

因为  $R[\text{eax}] = 0000\ 00\text{A}0\text{H}$ ,  $R[\text{ecx}] = 0000\ 00\text{0}3\text{H}$ , 所以,  $R[\text{al}] = \text{A}0\text{H}$ ,  $[\text{cl}] = \text{0}3\text{H}$ 。

执行指令 "imulb % cl" 后,  $R[\text{ax}] = \text{FEE}0\text{H}$ , 即十进制数 -288。

因为高 9 位乘积不为全 0 或全 1, 故 CF 和 OF 标志全为 1。

验证：此例中 mulb 指令执行的运算是  $160 \times 3 = 480$ ,

而 imulb 指令执行的运算是  $-96 \times 3 = -288$ 。

扩展：有符号数乘法中：结果的高 9 位 (ah 的 8 位 + 符号位扩展的 1 位)

若高 9 位不全为相同值 (全 0 或 全 1), 说明低 8 位 (al) 存不下补码结果 (补码需要符号扩展才能正确表示), 此时 CF 和 OF 置为 1;

若高 9 位全为 0 或 全为 1, 说明低 8 位能存下结果, CF 和 OF 置为 0。

# 六、分析设计题（每小题10分）

扩展一下：

无符号数乘法中：  $R[a1] = A0H$ ,  $[c1] = 03H$

①  $A0H$  转十进制是： $10 \times 16^1 + 0 \times 16^0 = 160$ ,  $03H$  转十进制是： $0 \times 16^1 + 3 \times 16^0 = 3$

$$R[a1] \times R[c1] = 160 \times 3 = 480$$

② 十进制 480 转十六进制： $480 \div 16 = 30$  余  $0$  ,  $30 \div 16 = 1$  余  $14$  ( $E$ ) ,  $1 \div 16 = 0$  余  $1$  ,

所以结果是  $1E0H$ , 即：  $01E0H$

有符号数乘法中：  $R[a1] = A0H$ ,  $[c1] = 03H$

①  $A0H$  转二进制是：  $1010\ 0000$ , 0正1负, 反码 + 1 = -96

$03H$  转十进制是：  $0000\ 0101$ , 0正1负, 3

$$R[a1] \times R[c1] = -96 \times 3 = -288$$

② 十进制 -288 在计算机中以补码形式存储，转二进制：  $1000\ 0001\ 0010\ 0000$ ,

反码：  $1111\ 1110\ 1101\ 1111$ , 再 + 1, 为：  $1111\ 1110\ 1110\ 0000$ , 四合一转十六进制为：

$FEE0H$

# 六、分析设计题（每小题10分）

30. 某计算机的主存空间大小为 512MB,按字节编址。指令 cache 和数据 cache 分离,两种 cache 均有 16 个 cache 行,主存与 cache 交换的块大小为 128B,数据 cache 采用 2 路组相联、通写法和 LRU 替换算法。现有程序 P,其伪代码如题 30 图所示。

程序 P:

```
int a[ 128 ][ 128 ]
.....
int sum_array( )
{
    int i,j,sum = 0;
    for( i = 0; i < 128; i + + )
        for( j = 0; j < 128; j + + )
            sum + = a[ i ][ j ];
    return sum;
}
```

题 30 图

假定 i,j,sum 均分配在寄存器中,数组 a 按行优先方式存放,其首地址 640。试回答下列问题:

- (1) 数据 cache 的总容量(包括标记和有效位等)是多少字节?
- (2) 程序 P 的数据命中率是多少?

# 六、分析设计题（每小题10分）

答案：

(1) 解题思路：[明确以下四个概念]

- ① 确定**主存地址**位数；例如：主存空间大小为 512 MB，所以 $\log_2(512 \times 1024 \times 1024) = 29$ 位
- ② 确定**块内地址**位数；例如：主存块大小为 128 B，所以 $\log_2 128 = 7$ 位
- ③ 确定**cache组号**位数；例如：cache数据共 16 行，2路组相联，所以有  $16/2=8$ 组，所以 $\log_2 8 = 3$ 位
- ④ 确定**标记位数**；①-②-③ = 19位

每个cache行除用于存放**主存块**外，还有**有效位**、**标记**以及**修改位**和**使用位**（如LRU位）等控制位。

由题可知用**通写法**，**无须修改位**，那就剩下**有效位**、**标记**和**使用位**（如LRU位）

所以cache的总容量为  $(128 \times 8 + 19 + 1 + 1) \times 16 = 16270$ 位 = 2090B

说明：（**主存块** + **标记** + **有效位** + **使用位**）

**备注**：2路组相联，所以**使用位**为 $\log_2 2 = 1$ 位

4路组相联，所以**使用位**为 $\log_2 4 = 2$ 位

# 六、分析设计题（每小题10分）

(2)

由题可知，假定编译时 i、j、sum 均分配在寄存器中，故数据访问命中率仅需要考虑数组 a 的访问情况。按行优先方式存放，所以程序 P 中数组访问顺序与存放顺序相同，故依次访问的数组元素位于相邻单元；一个数组元素[int类型]所占的字节数是 4B

方法一：

程序共访问  $128 \times 128 = 2^7 \times 2^7 = 2^{4+10} = 16K$  次，所以 占  $16K \times 4B / 128B = 0.5K$  个主存块，

因为首地址  $640 / 128 = 5$ ，正好位于一个主存块的起始位置，故每次将一个主存块装入 cache 时，  
总是第一个数组元素缺失，其他命中，共缺失  $0.5K$  次，

所以数据访问的命中率为  $(16K - 0.5K) / 16K = 96.875\%$ 。

方法二：

每个内存块存放的元素： $128B / 4B = 32$  个元素

因为首地址  $640 / 128 = 5$ ，正好位于一个主存块的起始位置，故每次将一个主存块装入 cache 时，  
总是第一个数组元素缺失，其他命中。所以数据访问的命中率为  $(32-1) / 32 = 96.875\%$

谢谢大家