**### 解释题**

**1. \*\*计算机多层结构\*\***

**- 计算机系统通常由多个层次组成，包括硬件、操作系统、编译器和应用程序等。理解这些层次之间的关系和交互是基础。**

**2. \*\*cache的作用\*\***

**- Cache 是一种高速存储器，用于减少访问主存的时间。它通过存储最近或最常用的指令和数据来提高程序的执行速度。**

**3. \*\*CRC码\*\***

**- CRC（循环冗余校验）是一种错误检测机制，用于检测数据传输中的错误。它通过生成一个校验码并将其附加到数据中，接收方可以通过重新计算校验码来检测错误。**

**4. \*\*机器指令的二要素，微指令的二要素，条件转移和顺序执行的情况\*\***

**- 机器指令通常包含操作码和操作数两部分。**

**- 微指令是控制单元的一部分，通常包含操作码和地址两部分。**

**- 条件转移和顺序执行是控制流的重要组成部分，条件转移根据条件决定是否跳转到其他指令，顺序执行则是按顺序执行指令。**

**5. \*\*浮点数的规格化和溢出\*\***

**- 浮点数的规格化是指将浮点数表示为标准形式的过程。**

**- 溢出是指数值超过了表示范围，导致结果不准确。**

**### 计算题（40分）**

**1. \*\*-33的原码，反码，补码，规格化浮点数\*\***

**- 原码：直接表示负数的方法。**

**- 反码：将原码的每一位取反。**

**- 补码：在反码的基础上加1。**

**- 规格化浮点数：将浮点数表示为标准形式。**

**2. \*\*比较两个单精度浮点数的大小\*\***

**- 单精度浮点数的比较需要考虑其符号位、指数位和尾数位。**

**3. \*\*已经知道二地址指令109条，三地址指令30条，每个地址6位，求指令字长\*\***

**- 指令字长 = 地址位数 × 地址数量 + 操作码位数。**

**- 例如，二地址指令的字长 = 6 × 2 + 操作码位数。**

**4. \*\*指令互斥\*\***

**- 指令互斥是指在同一时间只能执行一条指令。**

**5. \*\*求x=-7和x=-5的布斯定理算法\*\***

**- 布斯定理算法是一种快速乘法算法，适用于二进制数的乘法运算。**

**### 简答题（30分）**

**1. \*\*微指令\*\***

**- 微指令是控制单元的一部分，用于实现指令的执行。**

**- 取指令需要的微指令条数取决于指令的复杂度。**

**- SR+DR——>ADDR微指令表示源寄存器和目的寄存器之间的数据传输。**

**2. \*\*命中率——原题\*\***

**- 命中率是指缓存中找到所需数据的概率。**

**3. \*\*字位拓展——原题\*\***

**- 字位拓展是指增加数据宽度以提高处理能力。**

**### 应该复习的知识点**

**1. \*\*计算机多层结构\*\***

**- 计算机系统的层次结构及其相互作用。**

**2. \*\*Cache 的作用\*\***

**- Cache 的工作原理及其对性能的影响。**

**3. \*\*CRC 码\*\***

**- CRC 的生成和校验过程。**

**4. \*\*机器指令和微指令\*\***

**- 机器指令和微指令的结构及其执行过程。**

**5. \*\*浮点数的规格化和溢出\*\***

**- 浮点数的表示方法及其规格化过程。**

**6. \*\*原码、反码、补码\*\***

**- 数值的不同编码方式及其转换过程。**

**7. \*\*单精度浮点数的比较\*\***

**- 浮点数的比较方法。**

**8. \*\*指令字长的计算\*\***

**- 指令字长的计算公式及其应用。**

**9. \*\*指令互斥\*\***

**- 指令互斥的概念及其影响。**

**10. \*\*布斯定理算法\*\***

**- 布斯定理算法的原理及其应用。**

**11. \*\*微指令\*\***

**- 微指令的结构及其执行过程。**

**12. \*\*命中率\*\***

**- 缓存命中率的计算及其影响。**

**13. \*\*字位拓展\*\***

**- 数据宽度扩展的原理及其应用。**

**通过复习这些知识点，可以更好地理解和应对考试中的相关问题。**