

# 关于小班研讨题的说明

本学期对以往的“小班回课”环节进行了优化，采用了“小班研讨题”的形式作为替代。**这并非完全颠覆原有方式，而是希望通过内容更聚焦、参与研讨人数更多的方式，帮助学生掌握课程内容的重点。**

以往的回课形式，在小班有限的时间内（如每人 20 至 30 分钟），学生往往较难正确的梳理回顾两个小时的大课内容，会出现流水账形式、重点不突出，或者以偏概全、遗漏了重点。因此，这次将大课中应重点学习掌握的核心内容提炼成研讨题，方便在小班课上集中讨论。

**需要注意，研讨题不是作业题，而更像是用问题的形式列出了重点内容的提纲。**研讨题是紧密结合课件和教材内容，如无特别说明是拓展研讨，是不需要用到超出课堂的扩展知识。学生基于研讨题的脉络，通过分析与讲解，达到巩固知识、深化理解、提升思考的目的。“拓展研讨”是否在小班上做，根据时间安排，**由小班老师决定。**

**除了研讨题，每次大课还会布置几道作业题。**由于都是基于本讲课程，部分作业题的内容，会和研讨题有关联。这些作业题，可以在课后时间完成，由助教批改，并在小班上讲解；也可以安排在小班上由学生上台分析、大家研讨。具体形式**由小班教师决定。**

## ICS 小班研讨题和作业题（第 2 讲）

注：研讨题的默认前提基础是本课程尤其是本讲的课件和对应教材内容，不一定每次都做强调说明，例如基于 x86 体系结构、Linux 操作系统、C 语言程序等。

### 研讨题 1：关于“位”（bit）和字节（byte）

- （1）计算机内部选用了二进制表达形式，那为什么要用十六进制？在什么场合使用？和十进制相比有什么优缺点？
- （2）参照课件第 4 页，设计几个 8 位的二进制数（不能用课件上现成的），手工转换成十进制和十六进制，按步骤展示过程，体现转换的速度差异。注意要多用到十六进制中字母的表示，熟悉十六进制的特点。
- （3）**拓展研讨：**现在主流的设计方案是 8 位为 1 个字节，4 个字节为 1 个字（int 整型数）。你觉得这样设计是什么原因？有什么优点和不足？还有其他更好或者也很好的方式吗？
- （4）参照课件第 8 页，设计两个单字节的二进制数（不能用课件上现成的），展示位级的与、或、非、异或运算。
- （5）参照课件第 11 页，设计几个单字节的数（十六进制表达形式，不能用课件上现成的），展示 C 语言中逻辑操作的与、或、非。比较分析这些操作和位级操作的异同。
- （6）课件第 11 页提到了“p&&\*p”可以避免空指针访问，找出现实中关于这种操作的实际代码片段，并说明其原理。
- （7）参照课件第 12 页，设计几个单字节的二进制数（不能用课件上现成的），展示左移和两种右移操作。结合本讲后面学到的知识，分析为什么需要两种右移。

### 研讨题 2：关于整型数

- （1）参照课件第 14 页，设计几个 short int 类型的数（不能用课件上现成的），用 B2U 和 B2T 两个公式分别进行十进制到二进制的手工转换，按步骤展示过程。要包含正数、负数和最大最小等边界情况。
- （2）教材上提到了有符号数的其他编码方法（原码、反码），介绍其编码规则，并和补码比较优缺点。
- （3）**拓展研讨：**课件第 19 页提到的补码数取反操作，证明其成立。
- （4）参照课件第 14 页，设计几个有符号数和无符号数（不能用课件上现成的），分析它们进行比较时成立的关系。
- （5）简述扩展和截断的原则，重点分析为什么要用符号扩展方式（相比于直接在高位补 0），有什么用途。可以结合后面的乘法和除法等运算来思考。

- (6) 设计几个数，分别在无符号数加法中产生溢出、在有符号数加法中产生两种溢出。
- (7) 基于课件第 48 页，分析为什么会用移位来代替部分乘法操作，分析哪些乘法操作不会被移位代替。
- (8) 参照课件第 61 页，设计几个 4 字节的数（不能用课件上现成的），分别用大端模式和小端模式放在某个指定的内存位置。
- (9) **拓展研讨：**如果一台基于 x86（采用小端格式）的计算机从互联网上获取了一些数据（采用大端格式），那会不会对这些数据进行格式转换？如果转换，在什么地方、在什么时候转换？

## 作业题

第 2 讲 bit byte int 的作业题，在教材第 2 章，页码和题号如下：

- P88 2.59 编写一个 C 表达式，……
- P88 2.60 假设我们将一个……
- P91 2.71 你刚刚开始……