

作业的设置初衷是为了让同学们更好的发现问题，因此分数只是作业完成情况的一个参考，评分标准如下：全对 100 分，错 1 题 95 分，错 2 题 90 分，错 3 题以上 80 分

Amdahl 定律

第一题：

假设在一段程序中，整数加法指令出现的频率为 30%，其 CPI 为 5，浮点平方根指令出现的频率为 2%，其 CPI 为 30，其他指令的平均 CPI 为 1.25。

现有两种改进方案，方案 A 能够使得整数加法指令的 CPI 下降为 3，方案 B 能够使得浮点平方根指令的 CPI 下降为 20，试比较两种优化方案的加速比。

常见问题：

1. 需注意加速比的定义是程序执行时间的反比，因此直接将 CPI 的比值作为加速比是不严谨的，只是因为本题中没有针对 IC 和 C 的优化，所以最终的加速比就是 CPI 的比值。这一点提醒同学们注意，不过没有考虑 IC 和 C 的同学也没有扣分。
2. 本题一个常见的错误在于，Amdahl 定律中的 F_p 是指某一指令的执行占总执行时间的比例，而不是指令出现的频率，因此不能直接将本题中的 30% 直接代入 Amdahl 定律进行计算。

解：

改进前 $CPI = 5 \times 0.3 + 30 \times 0.02 + 1.25 \times 0.68 = 2.95$

方案 A $CPI = 3 \times 0.3 + 30 \times 0.02 + 1.25 \times 0.68 = 2.35$

方案 B $CPI = 5 \times 0.3 + 20 \times 0.02 + 1.25 \times 0.68 = 2.75$

设总运行时间为 T，指令条数为 IC，时钟周期为 C 则有

$$S_A = \frac{T}{T_A} = \frac{CPI * IC * C}{CPI_A * IC * C} = \frac{CPI}{CPI_A} \approx 1.26$$
$$S_B = \frac{T}{T_B} = \frac{CPI * IC * C}{CPI_B * IC * C} = \frac{CPI}{CPI_B} \approx 1.07$$

方案 A 的加速比更高

第二题：

假定要将某计算机系统的某一执行部件速度提高 10 倍，且已知改进后被改进部件的执行时间占系统总运行时间的 50%，试问改进后，整个系统获得的加速比是多少？

常见问题：

1. 本题题面的“提高 10 倍”描述不够规范，可以理解为加速比为 10 倍，也可以理解为加速比为 11 倍，相对应的答案有 5.5 和 6 两种，都被认为是正确的。
2. 本题常见的错误在于将“改进后”看作“改进前”。

解：

设改进前被改进部件的执行时间占系统总运行时间的 x，则由于改进后改部件占系统总运行时间的 50%，有

$$\frac{\frac{x}{10}}{1-x+\frac{x}{10}} = 50\%$$

可得 $x = \frac{10}{11}$

则有 $F_e = \frac{10}{11}$, $S_e = 10$, 根据 Amdahl 定律得到:

$$S_n = \frac{1}{1 - F_e + \frac{F_e}{S_e}} = 5.5$$

因此整个系统的加速比为 5.5

Huffman 编码

假设一台计算机共有 9 种不同的操作码, 已知各种操作码在程序中出现的概率如下表所示:

指令序号	I1	I2	I3	I4	I5	I6	I7	I8	I9
出现概率	0.43	0.13	0.07	0.06	0.05	0.01	0.02	0.22	0.01

请计算 Huffman 码、3/3/3 扩展码、定长操作码对应的平均码长和信息冗余量。

常见问题:

1. 在构造 Huffman 树时, 每一次是将出现概率最小的两棵子树合并。在本题的构造过程中开始的几个较小的概率合并成概率为 0.09 的一棵子树, 此时还尚未合并的有概率 0.06 的子树 I4 和概率 0.07 的子树 I3, 一种常见的错误是将 0.09 的子树和 0.06 的子树合并, 这样算出的答案是 2.44, 但正确的做法应当是将 0.06 和 0.07 的子树合并, 这样的答案为 2.42
2. 需要注意 3/3/3 扩展码的扩展方式
3. 同学们信息冗余量计算和标准答案可能有一点点偏差, 由于本课程并不考察计算, 因此只要式子正确就算正确。

解:

理论最短码长: $-\sum_{i=1}^9 p_i * \log_2 p_i = 2.36$

Huffman 树可能不止一棵, 但是平均码长是确定的, 一种可能的 Huffman 编码如下:

指令序号	I1	I2	I3	I4	I5	I6	I7	I8	I9
出现概率	0	110	11100	11101	11110	1111111	1111110	10	1111110

Huffman 编码的平均码长为 2.42, 信息冗余量为 $\frac{2.42-2.36}{2.42} = 2.48\%$

3/3/3 扩展码如下:

指令序号	I1	I2	I3	I4	I5	I6	I7	I8	I9
出现概率	00	01	10	1100	1101	1110	111100	111101	111110

它的平均码长为 2.52, 信息冗余量为 $\frac{2.52-2.36}{2.52} = 6.35\%$

定长扩展码的平均码长为 4, 信息冗余量为 $\frac{4-2.36}{4} = 41\%$

地址码个数的选择

使用三地址码、二地址码、二地址寄存器型码、一地址码、零地址码指令计算如下算式的过程中, 对应的指令条数、访存次数、程序存储量、访存信息量分别是多少?

$$x = \frac{a - b}{c * d + f}$$

(使用 P 表示操作码长度, A 表示地址码长度, D 表示数据长度, R 表示通用寄存器的地址码长度, B 表示字节数, 可以假设 $D=2A=8P=16R=8B$)

常见问题:

1. 计算需谨慎, 很多同学出现了计算错误
2. 一地址码的最后一步需要将计算得到的结果 store 到 X
3. 部分同学没有给出二地址寄存器型码的结果

解:

地址码答案不唯一, 下列仅供参考

三地址码:

SUB X, A, B
MUL Y, C, D
ADD Y, Y, F
DIV X, X, Y

二地址码:

MOVE X, A
SUB X, B
MOVE Y, C
MUL Y, D
ADD Y, F
DIV X, Y

二地址寄存器码:

MOVE R1, A

SUB R1, B
 MOVE R2, C
 MUL R2,D
 ADD R2, F
 DIV R1, R2
 MOVE X, R1

一地址码:

LOAD C
 MUL D
 ADD F
 STORE X
 LOAD A
 SUB B
 DIV X
 STORE X

零地址码:

PUSH A
 PUSH B
 SUB
 PUSH C
 PUSH D
 MUL
 PUSH F
 ADD
 DIV
 POP

地址数目	指令条数	访存次数	程序存储量	访存信息量
三地址	4	$4+12=16$	$4P+12A=52B$	$4P+12A+12D=148B$
二地址	6	$6+16=22$	$6P+12A=54B$	$6P+12A+16D=182B$
一地址	8	$8+8=16$	$8P+8A=40B$	$8P+8A+8D=104B$
零地址	10	$10+24=34$	$10P+6A=34B$	$10P+6A+24D=226B$
二地址寄存器 型	7	$7+6=13$	$7P+8R+6A=35B$	$7P+8R+6A+6D=83B$