计算机体系结构第 1 次作业

学号：2017218007　　姓名：文华　　班级：物联网工程 17-2 班

1.13　[10/20/20] <1.9> Your company is trying to choose between purchasing the Opteron or Itanium 2. You have analyzed your company's applications, and 60% of the time it will be running applications similar to wupwise, 20% of the time applications similar to ammp, and 20% of the time applications similar to apsi.

a. [10] If you were choosing just based on overall SPEC performance, which would you choose and why?
b. [20] What is the weighted average of execution time ratios for this mix of applications for the Opteron and Itanium 2?
c. [20] What is the speedup of the Opteron over the Itanium 2?

解：

a.

∵Itanium/Opteron

因为 Opteron 的 wupwise，　ammp 和 apsi 分别为 51.5、136.0 与 150.0 秒，且从题设可知该公司的程序时间占比为 wupwise 60%，ammp 20%及 apsi 20%，因此 Opteron 的 SPEC 指标计算如下（设总时间为 N）：

$SPEC_{Opteron}$ = 0.6N × 51.5 + 0.2N × 136.0 + 0.2N × 150.0 = 88.1N.

同理可计算 Itanium 2 的 SPEC：

$SPEC_{Itanium2}$ = 0.6N × 56.1 + 0.2N × 132 + 0.2N × 231.0 = 106.26N.

又∵

$$\frac{SPECRadio_A}{SPECRadio_B} = \frac{Performance_A}{Performance_B}$$

显然对于该公司的程序 Itanium 2 更合适.

∴选 Itanium 2.

b. Opteron: $0.6 \times 0.92 + 0.2 \times 1.03 + 0.2 \times 0.65 = 0.888$

c.

$$Speeup = \frac{T_{Itnium\ 2}}{T_{Optreon}} = \frac{1}{0.888} = \frac{125}{111}$$

1.14 [20/10/10/10/15] <1.9> In this exercise, assume that we are considering enhancing a machine by adding vector hardware to it. When a computation is run in vector mode on the vector hardware, it is 10 times faster than the normal mode of execution. We call the percentage of time that could be spent using vector mode the percentage of vectorization. Vectors are discussed in Chapter 4, but you don’t need to know anything about how they work to answer this question!

a. [20] <1.9> Draw a graph that plots the speedup as a percentage of the computation performed in vector mode. Label the y-axis “Net speedup” and label the x-axis “Percent vectorization.”

b. [10] <1.9> What percentage of vectorization is needed to achieve a speedup of 2?

c. [10] <1.9> What percentage of the computation run time is spent in vector mode if a speedup of 2 is achieved?

d. [10] <1.9> What percentage of vectorization is needed to achieve one-half the maximum speedup attainable from using vector mode?

e. [15] <1.9> Suppose you have measured the percentage of vectorization of the program to be 70%. The hardware design group estimates it can speed up the vector hardware even more with significant additional investment. You wonder whether the compiler crew could increase the percentage of vectorization, instead. What percentage of vectorization would the compiler team need to achieve in order to equal an addition 2× speedup in the vector unit (beyond the initial 10×)?

解：

a. 有以下结论：

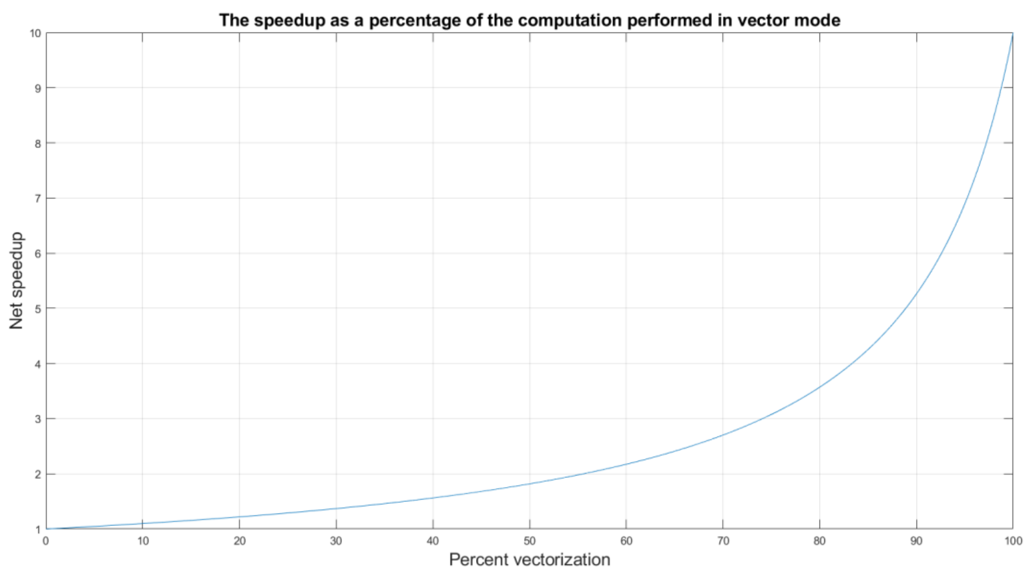
The overall speedup is the ratio of the execution times:

$$Speedup_{overall} = \frac{Execution\ time_{old}}{Execution\ time_{new}} = \frac{1}{1 - Fraction_{enhanced} + \frac{Fraction_{enhanced}}{Speedup_{enhanced}}}$$

根据题设知：在向量硬件上运行向量模式可以加速 10 倍，因此得如下拟合公式：

$$f(x) = \frac{100}{100 - x + \frac{x}{10}}$$

使用 matlab 绘制曲线如下图所示：



b.

设向量化占比 x，则可以列出如下方程：

$$\frac{1}{1 - x + \frac{x}{10}} = 100$$

解得：x = 0.56.

c.

$$\because\ Speedup_{new} = \frac{Time_{new}}{Time_{old}}$$

$\therefore \frac{0.056}{0.5} = 0.112$

d.

最大向量化率为：

$$\lim_{x \to \infty} \frac{100}{100 - x + \frac{x}{10}} = 10$$

$$\frac{10}{2} = \frac{1}{1 - x + \frac{x}{10}}$$

解得：

$$x = \frac{8}{9}$$

e.

$$\frac{1}{1 - x + \frac{x}{10}} = \frac{1}{1 - 0.7 + \frac{0.7}{10}} = \frac{100}{37} \approx 2.7$$

$$2 \times 2.7 = 5.4$$

根据书上结论列方程：

$$\frac{1}{1 - x + \frac{x}{10}} = 5.4$$

解得：$x = 0.91$

1.16 [20/20/15] <1.9> When making changes to optimize part of a processor, it is often the case that speeding up one type of instruction comes at the cost of slowing down something else. For example, if we put in a complicated fast floatingpoint unit, that takes space, and omething might have to be moved farther away from the middle to accommodate it, adding an extra cycle in delay to reach that unit. The basic Amdahl’s law equation does not take into account this trade-off.

a. [20] <1.9> If the new fast floating-point unit speeds up floating-point operations by, on average, 2×, and floating-point operations take 20% of the original program’s execution time, what is the overall speedup (ignoring the penalty to any other instructions)?

b. [20] <1.9> Now assume that speeding up the floating-point unit slowed down data cache accesses, resulting in a 1.5× slowdown (or 2/3 speedup). Data cache accesses consume 10% of the execution time. What is the overall speedup now?

c. [15] <1.9> After implementing the new floating-point operations, what percentage of execution time is spent on floating-point operations? What percentage is spent on data cache accesses?

解：

a. 根据题设与结论：

$$Speedup_{new\ FP} = \frac{CPU\ time_{original}}{CPU\ time_{new\ FP}} = \frac{1}{0.8 + \frac{0.2}{2}} = \frac{10}{9} \approx 1.11$$

b. 根据题设与结论：

$$Speedup_{overall} = \frac{CPU\ time_{original}}{CPU\ time_{new\ FP}} = \frac{1}{1 - 0.3 + \frac{0.2}{2} + 0.1 \times \frac{3}{2}} = 1.05$$

c.

$$\because \ \ Speedup_{new} = \frac{Time_{new}}{Time_{old}}$$

对于浮点操作：

$$\frac{0.1}{1 - \frac{0.2}{2}} = 0.105$$

对于 cache 访存：

$$\frac{0.15}{1 - \frac{0.2}{2}} \approx 0.158$$

1.18 [10/20/20/20/25] <1.10> When parallelizing an application, the ideal speedup is speeding up by the number of processors. This is limited by two things: percentage of the application that can be parallelized and the cost of communication. Amdahl’s law takes into account the former but not the latter.

a. [10] <1.10> What is the speedup with N processors if 80% of the application is parallelizable, ignoring the cost of communication?

b. [20] <1.10> What is the speedup with 8 processors if, for every processor added, the communication overhead is 0.5% of the original execution time.

c. [20] <1.10> What is the speedup with 8 processors if, for every time the number of processors is doubled, the communication overhead is increased by 0.5% of the original execution time?

d. [20] <1.10> What is the speedup with N processors if, for every time the number of processors is doubled, the communication overhead is increased by 0.5% of the original execution time?

e. [25] <1.10> Write the general equation that solves this question: What is the number of processors with the highest speedup in an application in which P% of the original execution time is parallelizable, and, for every time the number of processors is doubled, the communication is increased by 0.5% of the original execution time?

解：

a.

根据结论：

$$Speedup_{overall} = \frac{Execution\ time_{old}}{Execution\ time_{new}} = \frac{1}{1 + \frac{Fraction_{enhanced}}{Speedup_{enhanced}}} = \frac{1}{1 - 0.8 + \frac{0.2}{N}}$$

b.

$$\frac{1}{0.2 + 8 \times 0.005 + \frac{0.8}{8}} = \frac{1}{0.34} \approx 2.94$$

c.

$$\frac{1}{0.2 + 3 \times 0.005 + \frac{0.8}{8}} = \frac{1}{0.315} \approx 3.17$$

d.

$$Speedup_{overall} = \frac{Execution\ time_{old}}{Execution\ time_{new}} = \frac{1}{0.2 + log_2 N \times 0.005 + \frac{0.8}{8}}$$

e.

$$Speedup_{overall} = \frac{Execution\ time_{old}}{Execution\ time_{new}} = \frac{1}{1 - Fraction_{enhanced} + \frac{Fraction_{enhanced}}{Speedup_{enhanced}}}$$

以处理器的数量 N 代入上式并求导使等式右端等于 0，即

$$\therefore \frac{d}{dN} \times \frac{1}{1 - P + log_2 N \times 0.005 + \frac{P}{N}} = 0$$

□