

Módulo 1

Avaliação do Desempenho: Métricas

Pretende-se com esta sessão prática que os alunos percebam quais as grandezas envolvidas na avaliação do desempenho de um processador e a forma como estas se relacionam.

Exercícios

- 1) Considere um programa P escrito em C que foi compilado com o compilador Cv1 para a arquitectura aGG, resultando num programa Pv1 em código binário que executa um total de 10^6 instruções.

- a) O processador pGG01a implementa a arquitectura aGG, apresenta um CPI de 2.5 para este programa Pv1 e tem uma frequência do relógio de 0.5 GHz. Qual o tempo de execução de Pv1?

$$T_{exec} = \#I * CPI / f = 10^6 * 2.5 / 0.5 * 10^9 = 5 * 10^{-3} s = 5 ms$$

- b) O processador pGG01b tem uma organização semelhante ao processador anterior, mas a frequência do relógio é de 0.75 GHz. Este aumento da frequência resulta num aumento do CPI para 3. Quantas vezes mais rápido é pGG01b do que pGG01a a executar Pv1?

$$ganho = \frac{T_{exec_pGG01a}}{T_{exec_pGG01b}} = \frac{\#I * CPI_{pGG01a} * f_{pGG01b}}{\#I * CPI_{pGG01b} * f_{pGG01a}} = \frac{2.5 * 0.75 * 10^9}{3.0 * 0.5 * 10^9} = 1.25$$

- c) O processador pGG02 usa uma organização muito mais complexa para implementar a arquitectura aGG, resultando numa frequência de 2.5 GHz e executando Pv1 em 2 ms. Qual o CPI?

$$T_{exec} = \#I * CPI / f \Leftrightarrow CPI = T_{exec} * f / \#I = 2 * 10^{-3} * 2.5 * 10^9 / 10^6 = 5$$

- d) Finalmente, foi desenvolvido uma nova versão do compilador, Cv2, que resulta num programa Pv2 com CPI=7.5 no processador pGG02, mas que executa apenas metade das instruções de Pv1. Quantas vezes é Pv2 mais rápido do que Pv1 neste processador?

$$ganho_{v2 \rightarrow v1} = \frac{T_{exec_pv1}}{T_{exec_pv2}} = \frac{\#I_{pv1} * CPI_{pv1} * f_{pGG02}}{\#I_{pv2} * CPI_{pv2} * f_{pGG02}} = \frac{2 * \#I_{pv2} * CPI_{pv1}}{\#I_{pv2} * CPI_{pv2}} = \frac{2 * 5}{7.5} = 1.33$$

2) Considere um processador com 3 classes de instruções com o CPI indicado na tabela 1:

Tipo de instrução	CPI
A	1
B	2
C	3

Tabela 1 – CPI por classe de instrução

Um programador tem que seleccionar o compilador a usar para uma determinada aplicação. O número de instruções de cada classe gerado por cada um dos compiladores é apresentado na tabela 2:

Compilador	Nº de instruções por classe		
	A	B	C
C1	$1 \cdot 10^6$	$3 \cdot 10^6$	$4 \cdot 10^6$
C2	$5 \cdot 10^6$	$2 \cdot 10^6$	$3 \cdot 10^6$

Tabela 2 – Nº de instruções por classe e compilador

- a) Calcule, para cada um dos compiladores o nº de instruções executadas, o CPI global e o número de *clock cycles* necessário para executar a aplicação. Qual o compilador que produz código mais rápido? Quantas vezes mais rápido que o outro compilador?

$$\#I_{C1} = 8 \cdot 10^6$$

$$\#I_{C2} = 10^7$$

$$CPI_{C1} = \frac{1 \cdot 10^6 + 2 \cdot 3 \cdot 10^6 + 3 \cdot 4 \cdot 10^6}{8 \cdot 10^6} = \frac{19}{8}$$

$$CPI_{C2} = \frac{5 \cdot 10^6 + 2 \cdot 2 \cdot 10^6 + 3 \cdot 3 \cdot 10^6}{10^7} = 1.8$$

$$CC_{C1} = \#I_{C1} \cdot CPI_{C1} = 19 \cdot 10^6$$

$$CC_{C2} = \#I_{C2} \cdot CPI_{C2} = 18 \cdot 10^6$$

- b) Se a frequência de relógio deste processador é de 1 GHz, qual o tempo de execução da aplicação?

$$T_{exec_C1} = 19 \cdot 10^6 / 10^9 = 19 \text{ ms}$$

$$T_{exec_C2} = 18 \cdot 10^6 / 10^9 = 18 \text{ ms}$$

- c) Com algumas alterações da organização do processador e da tecnologia usada para o construir, uma equipa de projetistas conseguiu aumentar a frequência do relógio para 2 GHz. No entanto, o CPI de cada uma das classes de instruções aumentou, conforme ilustrado na tabela 3. Calcule o tempo de execução do código gerado por cada um dos compiladores. Qual é agora o compilador que gera código mais rápido?

Tipo de instrução	CPI
A	2
B	3
C	4

Tabela 3 – CPI por classe de instrução

$$CC_{C1} = (2 + 3 \cdot 3 + 4 \cdot 4)10^6 = 27 \cdot 10^6$$

$$CC_{C2} = (2 \cdot 5 + 3 \cdot 2 + 4 \cdot 3)10^6 = 28 \cdot 10^6$$

$$T_{exec_C1} = 27 \cdot 10^6 / 2 \cdot 10^9 = 13.5 \text{ ms}$$

$$T_{exec_C2} = 28 \cdot 10^6 / 2 \cdot 10^9 = 14 \text{ ms}$$

- 3) Considere que o projecto de um processador e respetivo compilador está pronto. Compete-lhe agora decidir se este projeto deve ser implementado, ou se deve ser investido mais tempo a melhorá-lo. As opções disponíveis são:
- Manter o projeto como está – frequência do relógio de 1.5 GHz e com o CPI por classe de instrução e percentagem de ocorrência de cada classe conforme apresentado na tabela 4. Esta opção é designada por *base*.
 - Introduzir alterações na organização do processador - frequência do relógio de 2 GHz e com o CPI por classe de instrução e frequência de ocorrência de cada classe conforme apresentado na tabela 4. Esta opção é designada por *hard*.

Classe	CPI _{base}	CPI _{hard}	% Ocorrência
A	2	2	40%
B	3	2	25%
C	3	3	25%
D	5	4	10%

Tabela 4 – Dados para opção *base* e *hard*

- a) Qual o CPI médio de cada uma das máquinas?

$$CPI_{base} = 2 * 0.4 + 3 * .25 + 3 * .25 + 5 * .1 = 2.8$$

$$CPI_{hard} = 2 * 0.4 + 2 * .25 + 3 * .25 + 4 * .1 = 2.45$$

- b) **NOTA: é provável que a métrica MIPS ainda não tenha sido introduzida nas aulas T para alguns turnos.** Lembrando que o MIPS nativo é dado por $MIPS = \#I / (T_{exec} * 10^6)$, qual o MIPS nativo de cada uma das máquinas? E o MIPS de pico (peak)?

$$MIPS = \frac{\#I}{T_{exec} * 10^6} = \frac{\#I * f}{\#I * CPI * 10^6} = \frac{f}{CPI * 10^6} \Rightarrow MIPS_{base} = 536 \text{ e } MIPS_{hard} = 816$$

$$MIPS_{peak} = \frac{f}{CPI_{minimo} * 10^6} \Rightarrow MIPS_{peak_base} = \frac{1.5 * 10^9}{2 * 10^6} = 750 \text{ e } MIPS_{peak_hard} = \frac{2 * 10^9}{2 * 10^6} = 1000$$

- c) Qual o ganho em desempenho conseguido com o processador otimizado para um programa típico?

$$ganho_{hard \rightarrow base} = \frac{T_{exec_base}}{T_{exec_hard}} = \frac{\#I * CPI_{base} * f_{hard}}{\#I * CPI_{hard} * f_{base}} = \frac{2.8 * 2 * 10^9}{2.45 * 1.5 * 10^9} = 1.52$$

- d) A equipa responsável pelo compilador pensa conseguir desenvolver um produto mais sofisticado em que as percentagens de instruções de cada tipo são dadas pelos valores indicados na tabela 5. Este compilador gera, no entanto, mais 10% de instruções do que o anterior. A opção, que consiste na máquina *hard* com o compilador otimizado, é designada por *opt*. Qual o CPI obtido ? Qual o ganho relativamente à versão *base* com o compilador original?

Classe	% Ocorrência
A	50%
B	25%
C	20%
D	5%

Tabela 5 – Percentagem de instruções executadas com a opção *opt*

$$CPI_{opt} = 2 * 0.5 + 2 * .25 + 3 * .2 + 4 * .05 = 2.3$$

$$ganho_{opt \rightarrow base} = \frac{T_{exec_base}}{T_{exec_opt}} = \frac{\#I * CPI_{base} * f_{hard}}{1.1 * \#I * CPI_{opt} * f_{base}} = \frac{2.8 * 2 * 10^9}{1.1 * 2.3 * 1.5 * 10^9} = 1.47$$

- 4) Considere o excerto de programa *assembly* abaixo e o CPI para cada tipo de instruções de acordo com a tabela 6.

```
ciclo:
    movl 10(%ebx, %ecx, 4), %edx
    addl %edx, %eax
    subl $2, %ecx
    jnz  ciclo
```

Tipo de Instrução	CPI
Acesso à memória	5
Saltos condicionais	1
Operações inteiras	2

Tabela 6 - CPI por tipo de instrução

- a) Qual o CPI global para este excerto de programa?

$$CPI = .25 * 5 + .25 * 1 + .5 * 2 = 2.5$$

- b) Sabendo que o valor inicial de %ecx é de 10000, quantos ciclos do relógio são necessários para executar este programa?

$$CC = \#I * CPI = \frac{10000}{2} * 4 * 2.5 = 50000 \text{ cc}$$

- c) Sabendo que o tempo de execução deste excerto de programa num determinado processador é de 20 microsegundos, qual a frequência do relógio?

$$T_{exec} = \#I * CPI / f \Leftrightarrow f = \#I * CPI / T_{exec} = 50000 / 20 * 10^{-6} = 2.5 \text{ GHz}$$