

1. Introdução

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

1.1. Conteúdos e Resultados de Aprendizagem relacionados

Conteúdos	7.1 – Tempo de execução	C1
	7.2 – Ciclos por Instrução (CPI)	C1
Resultados de Aprendizagem	R7.1 – Identificar e caracterizar as métricas relativas ao desempenho da execução de um programa	C1
	R7.2 – Utilizar modelos quantitativos para prever/avaliar o desempenho de uma máquina	C1

2. Exercícios

- 1) 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:

Nº instruções por classe			
Compilador	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

- 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?
- Se a frequência de relógio deste processador é de 1 GHz, qual o tempo de execução da aplicação?
- Com algumas alterações da organização do processador e da tecnologia usada para o construir, uma equipa de projectistas 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 de cada uma das sequências de código.

Tipo de instrução	CPI
A	2
B	3
C	4

Tabela 3 – CPI por classe de instrução

- 2) Considere que o projecto de um processador e respectivo compilador está pronto. Compete-lhe agora decidir se este projecto deve ser implementado, ou se deve ser investido mais tempo a melhorá-lo. As opções disponíveis são:
- Manter o projecto como está – frequência do relógio de 1.5 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 *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}	Freq
A	2	2	40%
B	3	2	25%
C	3	3	25%
D	5	4	10%

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

- Qual o CPI global de cada uma das máquinas?
- 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)?
- Qual o ganho em desempenho conseguido com o processador otimizado para um programa típico?
- A equipa responsável pelo compilador pensa conseguir desenvolver um produto mais sofisticado que reduziria o número de instruções nas proporções indicadas na tabela 5. A opção, que consiste na máquina base com o compilador otimizado, é designada por *comp*. Qual o CPI obtido ? Qual ganho relativamente a *base* e a *hard*?

Classe	% relativamente a <i>base</i>
A	90%
B	90%
C	85%
D	95%

Tabela 5 – Percentagem de instruções executadas com a opção *comp* relativamente a *base*

- A combinação da versão *hard* e *comp* é designada por *comb*. Qual o CPI global e qual o ganho de desempenho relativamente a *base*?
- Sabendo que o desempenho dos CPUs aumenta, em geral, aproximadamente 1.5 por ano (cerca de 1.034 ao mês), que o tempo necessário para desenvolver cada uma das opções é dado pela tabela 6 e que a versão *base* tem um desempenho idêntico ao dos seus concorrentes directos, quais as opções que valem a pena seguir tendo em atenção o desenvolvimento da concorrência?

Opção	Tempo para desenvolver
Hard	6 meses
Comp	6 meses
Comb	8 meses

Tabela 6 – Tempos de desenvolvimento de cada uma das opções

- 3) Considere a máquina FP que dispõe de uma unidade de vírgula flutuante, realizando estas operações em hardware. A tabela 7 indica a percentagem de ocorrência típica de cada tipo de instrução para um programa genérico e o respectivo CPI.

Instrução	% ocorrência	CPI
Multiplicação FP	10 %	6
Divisão FP	5 %	20
Adição FP	30 %	4
Operações inteiras	55 %	2

Tabela 7 - % de ocorrência e CPI para a máquina FP

- Qual o CPI global para este programa genérico?
- Suponha que é possível reduzir para metade o CPI das divisões FP. Qual o ganho de desempenho?
- Se em vez de reduzir o CPI das divisões fosse possível reduzir para metade o CPI das adições FP, qual seria o ganho de desempenho? Como interpreta este resultado à luz da lei de Amdahl?
- Suponha uma outra máquina NFP que não tem uma unidade de vírgula flutuante, realizando todos os cálculos usando a unidade de inteiros. Sabe-se que uma multiplicação FP requer 30 operações inteiras, uma adição FP requer 20 e uma divisão requer 50. Quantas vezes mais rápida é a máquina FP para o programa apresentado?
- Se ambas as máquinas tiverem uma frequência do relógio de 1 GHz qual o MIPS nativo de cada uma?