

1 – Desempenho

1. Considere três processadores, CPU1, CPU2 e CPU3, que executam o mesmo *instruction set*, e que para um determinado programa P, apresentam as seguintes características:
 - CPU1 tem uma frequência de 3 GHz e um CPI = 1.5;
 - CPU2 tem uma frequência de 2.5 GHz e um CPI = 1.0;
 - CPU3 tem uma frequência de 4.0 GHz e um CPI = 2.2.
 - a) Qual o processador que exibe melhor desempenho medido em milhões de instruções por segundo (MIPS)
 - b) Se o processador CPU1 executa o programa P em 10 segundos, qual o número de instruções deste programa?
 - c) Indique, **justificando**, qual o processador que executa em menor tempo o programa P.
2. Justifique as duas seguintes afirmações:
 - a) “O aumento da frequência de um processador, sem qualquer outra alteração na sua organização, resulta num aumento do CPI”.
 - b) “Considere que um programa expresso numa linguagem de alto nível é compilado 2 vezes, com e sem optimização. Se a versão sem optimização (P-O0) implicar a execução de mais instruções do que a versão optimizada (P-O2) então, na generalidade dos casos, a versão p-O0 apresenta menor CPI do que P-O2”

[illegible]

- [illegible]

3 – Hierarquia de Memória

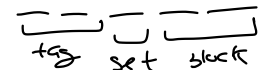
1. A tabela abaixo apresenta o estado de uma cache com um total de 4 linhas, B=4 e m=5. A L identifica a linha (em binário). A coluna *tag* apresenta o valor deste campo, com 2 bits, para as linhas cujo *valid* bit (coluna seguinte) esteja a 1. As 4 colunas seguintes apresentam, em hexadecimal, o valor de cada um dos *bytes* carregados na cache.

L	tag	valid	Bytes			
			00	01	10	11
00	11 10 = 00	1	0x23	0x7B	0xFF	0x00
01	--	0	0x23	0xBF	0x40	0x05
10	00 = 00	1	0x0F	0xAC	0xCD	0x10
11	11	1	0x12	0x05	0x8F	0xD0

A próxima tabela apresenta o conteúdo da memória:

Addr	Val	Addr	Val	Addr	Val	Addr	Val
00	0x23	08	0x01	16	0x23	24	0x22
01	0xBF	09	0x02	17	0x7B	25	0x55
02	0xA0	10	0xCD	18	0xFF	26	0xFF
03	0x05	11	0xB2	19	0x00	27	0x21
04	0x0F	12	0x02	20	0xC1	28	0x12
05	0xAC	13	0x23	21	0xD2	29	0x05
06	0xCD	14	0x9A	22	0xE3	30	0x8F
07	0x10	15	0xB4	23	0xB6	31	0xD0

- a) Indique qual a organização desta cache (S= 2 , E= 2 , B= 4 , m= 5).



- b) Considere um algoritmo de substituição LRU, durante a sequência de leitura de endereços de memória: 05, 03 e 26 (base 10). Para cada uma preencha os quadros, apenas com as alterações ao estado anterior. Indicando se trata de um **hit**, **cold miss** ou **colisão**, indicando a **linha** respetiva e ainda, o **conteúdo** da cache, sublinhando **valor do byte lido**, após cada acesso.

Endereço = 05 = 00101				Bytes			
Hit/cLM/colis	L	tag	valid	00	01	10	11
Hit	10	00					
Endereço = 03 = 00011				Bytes			
Hit/cLM/colis	L	tag	valid	00	01	10	11
Cold miss	01	00	1	0x23	0xBF	0x40	0x05
Endereço = 26 = 11010				Bytes			
Hit/cLM/colis	L	tag	valid	00	01	10	11
Colisão	00	11	1	0x22	0x55	0xFF	0x21

2. Uma dada máquina tem uma frequência do relógio de 2GHz e a respectiva cache apresenta uma *miss rate* de instruções de 2% e de dados de 5%. A miss penalty é de 20 nanosegundos. O CPI do CPU é dado pela tabela abaixo para diferentes tipos de instruções:

Tipo instrução	CPI _{CPU}
Acesso memória	1
Restantes	2

Para um valor de %ecx = 10000 um dos excertos do programa abaixo executa em 75 microsegundos. Indique, justificando, qual.

excerto1:

```
movl (%ebx), %eax 1
addl %eax, %esi    2
addl $4, %ebx      2
decl %ecx           2
jnz excerto1       2
```

excerto2:

```
movl %ebx, %eax
addl %eax, %esi
addl $4, %ebx
subl $2, %ecx
jnz excerto2
```

$$CPI_{CPU} = \frac{1+2+2+2+2}{5} = \frac{9}{5} = 1,8$$

$$CPI_{Mem} = (0,02 + \frac{1}{5} \cdot 0,05) \cdot 40 = 1,2$$

$$mp_T = 20 \text{ ns} = 2 \times 10^{-8} \text{ s} \rightarrow mp = 2 \times 10^{-8} \times 2 \times 10^9 = 40$$

$$T_{EXEC} = 50'000 \cdot 3 / (2 \times 10^9) = 90000075 = 7,5 \times 10^{-5} = 75 \times 10^{-6}$$

$$T_{EXEC} = 25'000 \cdot 3 / (2 \times 10^9) = 3,75 \times 10^{-5}$$

10'000 x 5 = 50'000
#I

25'000
#I

3 = CPI

$CPI_{CPU} = 2$
 $CPI_{Mem} = 0,8$

2,8
CPI

1 – Desempenho

1. Considere três processadores, CPU1, CPU2 e CPU3, que executam o mesmo *instruction set*, e que para um determinado programa P, apresentam as seguintes características:

- CPU1 tem uma frequência de 3 GHz e um CPI = 1.5;
 - CPU2 tem uma frequência de 2.5 GHz e um CPI = 1.0;
 - CPU3 tem uma frequência de 4.0 GHz e um CPI = 2.2.
- a) Qual o processador que exibe melhor desempenho medido em milhões de instruções por segundo (MIPS)
- b) Se o processador CPU1 executa o programa P em 10 segundos, qual o número de instruções deste programa?
- c) Indique, **justificando**, qual o processador que executa em menor tempo o programa P.

$$a) \text{MIPS} = \frac{\#I}{T_{\text{exec}} \cdot 10^6} = \frac{\#I}{\text{CPI} \cdot \#I / f \cdot 10^6}$$

$$\text{MIPS}_1 = \frac{3}{1.5 \cdot 10^6} = 2 \cdot 10^{-6}$$

$$\text{MIPS}_2 = \frac{2.5}{1 \cdot 10^6} = 2.5 \cdot 10^{-6}$$

$$\text{MIPS}_3 = \frac{4}{2.2 \cdot 10^6} = 1.8 \cdot 10^{-6}$$

$$b) T_{\text{exec}} = \text{CPI} \cdot \#I / f$$
$$10 = 1.5 \cdot x / 3 \cdot 10^9$$
$$x = 2 \cdot 10^{10}$$

$$c) T_{\text{exec}1} = 5 \cdot 10^9$$
$$T_{\text{exec}2} = 4 \cdot 10^8 \rightarrow \text{+ fast}$$
$$T_{\text{exec}3} = 5.5 \cdot 10^8$$

2. Justifique as duas seguintes afirmações:

- a) "O aumento da frequência de um processador, sem qualquer outra alteração na sua organização, resulta num aumento do CPI".
- b) "Considere que um programa expresso numa linguagem de alto nível é compilado 2 vezes, com e sem optimização. Se a versão sem optimização (P-O0) implicar a execução de mais instruções do que a versão optimizada (P-O2) então, na generalidade dos casos, a versão p-O0 apresenta menor CPI do que P-O2"

$$a) T_{\text{exec}} = \frac{\text{CPI} \cdot \#I}{f} \rightarrow \text{logo se a } f \text{ aumenta, o CPI também aumenta}$$

1. Considere que a lógica combinatória dos diversos estágios do *datapath* estudado nas aulas (4 estágios: F – *fetch*; D – *decode*; E – *execute* ; W – *writeback*) têm as seguintes latências apresentadas na tabela abaixo. Os registos a utilizar têm uma latência de 50 ps.

F	D	E	W
200 ps	250 ps	150 ps	350 ps

- Indique, **justificando**, quais as máximas frequências para um processador sem *pipeline* e um processador com *pipeline*;
- Como relaciona o ganho de desempenho com o *pipelining* com o número máximo de estágios que podem ser usados? Quais os factores que determinam que o ganho possa ser diferente do número de estágios?
- A versão sem *pipeline* tem um CPI=1. Já na versão com *pipeline* e para um programa genérico, 12.5% das instruções resultam em dependências de dados ou controlo que exigem em média 2 ciclos adicionais. Qual o ganho obtido com a versão com *pipeline* para este programa?

a) Sem *pipeline* $\rightarrow 1000 \text{ ps} = T_{cc} \parallel f = \frac{1}{1000 \times 10^{-12}} = 1 \text{ GHz}$
 Com *pipeline* e 4 estágios $\Rightarrow T_{cc} = 400 \text{ ps} \parallel f = \frac{1}{400 \times 10^{-12}} = 2,5 \text{ GHz}$

b) Usando *pipeline* a 'f' aumenta em relação ao não usar *pipeline*.
 $T_{exec} = \text{CPI} \cdot \#I / f$, logo se 'f' aumenta, o T_{exec} diminui.
 (Explicar o *fetch*, etc.)

c) $T_{exec \text{ com pipeline}} = (\text{CPI} \cdot \#I) / f \rightarrow \text{também aumenta} = 2,5 \cdot 10^9$
 $\rightarrow 0,875 \times 1 + 0,125 \times 2 = 1,125$

$$\text{ganho} = \frac{1 \cdot \#I / (1 \cdot 10^9)}{(1,125 \cdot \#I) / (2,5 \cdot 10^9)} = 2,2 \rightarrow \text{ganho de } 122\%$$

