

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]

- b) Considere que este processador resolve as dependências de dados através de *data forwarding*, isto é, realimenta valores do estágio E2 para o estágio D. Injectará ainda assim NOPs no estágio E1, mantendo a instrução dependente no estágio de *Decode* até que o *data forwarding* seja possível ou a dependência esteja resolvida. Preencha a tabela abaixo, indicando para cada ciclo do relógio qual a instrução ou bolha (NOP) que se encontra em cada estágio.

[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	10	1	0x23	0x7B	0xFF	0x00
01	--	0	--	--	--	--
10	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= , E= , B= , m=).
- 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				Bytes			
Hit/cLM/colis	L	tag	valid	00	01	10	11
Endereço = 03				Bytes			
Hit/cLM/colis	L	tag	valid	00	01	10	11
Endereço = 26				Bytes			
Hit/cLM/colis	L	tag	valid	00	01	10	11

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
addl %eax, %esi
addl $4, %ebx
decl %ecx
jnz excerto1
```

excerto2:

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