

**1** – Considere a seguinte sequência de código:

**1.1** – Indique quantos ciclos do relógio são necessários para executar este código na versão PIPE- do processador Y86. Justifique a sua resposta e use a tabela 1 para ilustrar a ocupação dos estágios do processador, sempre que tal lhe parecer apropriado.

```
.pos 0x0000F000
I1:  irmovl $2, %ecx
I2:  irmovl $0, %eax
I3:  irmovl $1, %esi
ciclo:
I4:  addl %ecx, %eax
I5:  subl %esi, %ecx
I6:  jne ciclo
I7:  halt
I8:  nop
```

**1.2** – Indique quantos ciclos do relógio são necessários para executar este código na versão PIPE do processador Y86. Justifique a sua resposta e use a tabela 2 para ilustrar a ocupação dos estágios do processador, sempre que tal lhe parecer apropriado.

**1.3** – Para as condições da última alínea indique qual o valor do PC no ciclo 7.

2 – Considere a seguinte sequência de código:

1	0x000: 8028000000		call	<b>proc</b>
2	0x005:		<b>fim:</b>	
3	0x005: 10		halt	
4	0x006:		<b>proc:</b>	
5	0x006: 90		ret	

Para a instrução assinalada como específica, preencha a tabela com os valores relevantes dos sinais de controlo, na versão **sequencial** do processador **Y86**

Estágios	Genéricos	Específica
	Instrução	ret
Extrai	icode:ifun $\leftarrow$ M <sub>1</sub> [PC] rA:rB $\leftarrow$ M...   ValC $\leftarrow$ M..   valP $\leftarrow$ PC...	
Descodifica	ValA $\leftarrow$ R[rA]   ValB $\leftarrow$ ...	
Executa	valE $\leftarrow$ ... ajusta CC   Bch $\leftarrow$ Cond(CC, ifun)	
Memória	M <sub>?</sub> $\leftarrow$   ValM $\leftarrow$	
Actualiza	R[?] $\leftarrow$ ...	
PC	PC $\leftarrow$ valP   PC $\leftarrow$ Cch: valC: ValP   ValM	

**3** – Suponha que se pretende acrescentar a instrução `ret<cond>` à arquitectura Y86. Esta instrução regressa de uma função, lendo o endereço de retorno do topo da pilha (*stack*) à semelhança do `ret`. Mas o salto apenas se dá se a condição for verdadeira. O leque de condições a suportar coincide com as previstas para os saltos condicionais.

**3.1** – Indique como codificaria esta instrução, de forma a compatibilizá-la com o formato usado pelas demais instruções do Y86.

**3.2** – Que tipo de dependências podem surgir com esta instrução numa organização com encadeamento e que penalizações implicam?

**3.3** – A técnica de previsão de destino dos saltos condicionais usada no Y86 é de estes são sempre tomados. Discuta a alternativa de salto não tomado para o `ret` condicional.

**Tabela 1**

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	Observações	

**Tabela 2**

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	Observações		

Número:

Nome: