

# ISA do IA-32 (parte 1)

## Teste 3

Nº	Nome
----	------

1. Num PC e num dado instante, o registo `%eax` contém o valor  $x$ , e o registo `%ecx` contém o valor  $y$ .

- a) (A) As 2 instruções seguintes têm algumas semelhanças (por ex., ambas ocupam 3 bytes), mas são distintas. Indique as principais diferenças.

```
movl 6(%eax), %edx
leal 6(%eax), %edx
```

- b) (A/R) Complete a tabela seguinte, com instruções e expressões compatíveis com as células já preenchidas; o registo `%edx` irá conter o resultado das instruções (valor).

Instrução	Valor
<code>movl 6(%eax), %edx</code>	$\text{Mem}[6 + x]$
	$x + y$
	$\text{Mem}[10 + 4y]$
	$-9 + 5y$
<code>leal(%eax,%ecx,4), %edx</code>	

2. Considere a seguinte expressão aritmética com inteiros, no corpo de uma função em C, para execução num PC (*little endian*),

$$a(i) = b + c$$

em que  $a$  é um elemento do único *array* local (cuja estrutura se encontra na memória, desde  $\text{Mem}[(\%ebp) - 80]$  até à célula contígua à apontada por `%ebp`),  $i$  é o índice do *array* que foi passado como argumento único da função (em  $\text{Mem}[(\%ebp) + 8]$ ), e  $b$  e  $c$  são variáveis locais (nos registos `%ecx` e `%esi`).

- a) (R) Indique, justificando, (i) a dimensão máxima deste *array* e (ii) uma expressão que permita calcular a localização de um elemento deste *array* (use a sintaxe da tabela anterior, e considere o índice do *array* em `%eax`).

Nº	Nome
----	------

b) (R/B) Considere que:

- o conteúdo de cada célula de memória é igual ao *byte* menos significativo do seu endereço;
- o valor de *i* já se encontra no registo `%eax`;
- os registos `%eax`, `%ecx`, `%esi`, e `%ebp` contêm, respectivamente, os seguintes valores em hexadecimal: `0x8`, `0xffffef`, `0x200`, `0x8401220`.

Mostre a localização e conteúdo das células que vão conter o valor de  $a(i)$  após o cálculo da expressão aritmética apresentada no início deste enunciado.

c) (B) Apresente o código *assembly* que poderia ser gerado pelo `gcc` para uma arquitectura típica RISC, quando encontrasse essa expressão (use a sintaxe do *assembler* do IA-32 da Gnu, e os pressupostos da alínea anterior).