

NOME:

Nº:

Notas:

1. Para cada uma das **5 questões de resposta satisfatória obrigatória**, numeradas de **1 a 5**, são-lhe oferecidas pelo menos 2 hipóteses para responder e/ou comentar; para cada um destas deverá optar por responder a **apenas por uma** delas.
3. Para cada uma das hipóteses que optar, deverá apresentar a **justificação da solução**, incluindo o raciocínio ou os cálculos que efectuar.
3. **Não são permitidas:** (i) máquinas de calcular e (ii) notas auxiliares de memória.
4. Correção de cada questão: não-satisfaz (**0**), satisfaz com erros (**0.8**), certa com falhas (**1.0**) e completamente certa (**1.2**).

1.

- a) Pretende-se armazenar numa variável o dia e o mês, relativos a uma data. Essa variável deve permitir saber-se que HOJE é o dia "10" do mês de "Maio". Sugira uma possível forma de codificar a informação dessa variável, indicando quantos bits são precisos.
- b) Um dispositivo móvel tem um ecrã quadrado com lado de 200 *pixels*. Cada *pixel* é representado por três inteiros (sem sinal), que indicam a intensidade de cada cor (*Red-Green-Blue*) nesse ponto. Diga quantos bits pode ter, no máximo, cada um desses inteiros, se a memória do ecrã tiver capacidade para 60 000 *bytes*.

2.

- a) Considere o seguinte fragmento de código em C:


```
int i=0;
if (i>=55)
    i++;
else
    i--;
```

Complete o mesmo fragmento compilado para *assembly* e introduza comentários.

```
xorl    _____, %ebx
cmpl   _____, %ebx
jle    .L2
_____ %ebx
jmp    .L3
.L2:
_____ %ebx
.L3:
```

- b) Represente a estrutura genérica de código C a que corresponde o seguinte fragmento de código *assembly*:

```
.L1:
cmpl   $0,%ebx
jge    .L2
...    # bloco de código
decl   %ebx
jmp    .L1
.L2: ...
```

Nota: Para as seguintes 3 questões escolha apenas uma afirmação, indique se **Verdadeira** ou **Falsa**, e justifique a sua resposta.

3. Considere um processador de **24 bits** (os inteiros são codificados em complemento para 2).

- a) Se num registo estiver o inteiro $0xc800$, então ele contém o valor -98k em decimal.
- b) Com este processador é possível representar como variável inteira uma distância em km até 1 minuto-luz (~18 milhões de km).

4. Considere uma versão reduzida da norma IEEE 754 com 10 bits (5 bits para o expoente em excesso de 15, 4 para a mantissa e 1 para o sinal; não esquecer os casos de excepção). O valor decimal de um n° normalizado representado com este formato vem dado por $V = (-1)^S * 1.F * 2^{(Exp-15)}$

- a) O operando 1100011010_2 representa o valor -6.5_{10}
- b) Esta versão reduzida a 10 bits não permite representar 2^{16}

5. Considere a execução duma instrução do IA-32 (*little endian*) representada em *assembly* por:

```
addl    %eax, -8(%ebp).
```

A instrução em binário está em memória, no endereço $0x8c20484$, e ocupa 3 células de memória: as 2 primeiras com o código da operação e a especificação do local onde estão os operandos, com os valores $0xbe$ e $0x7c$, enquanto a 3ª contém a constante -8 .

Considere os seguintes conteúdos de registos: $\%eax$ $0x8084$, $\%esp$ $0x8c28f00$, $\%ebp$ $0x8c28f0c$.

Considere ainda este conteúdo das 16 células no topo da *stack* (endereços por ordem crescente, em hexadecimal):
00, 00, 00, 00, 88, 00, 01, 00, 00, 00, 00, 00, 00, 00, 00, 00

- a) Por ordem cronológica, toda a informação que circula apenas no barramento de endereços é a seguinte:
 $0x08c20484$, $0x00008084$, $0x08c28f04$, $0x8c28f00$
- b) Após a execução desta instrução, a célula de memória no endereço $0x8c28f05$ foi alterada para 81.