

SISTEMAS COMPUTACIONAIS

LICENCIATURA EM ENGENHARIA INFORMÁTICA - 2º ANO

Cotações

- 1) _ 1.25
- 2) _ 1.25
- 3) _ 1.25
- 4) _ 1.25
- 5) _ 1.25
- 6) _ 1.25
- 7) _ 1.25
- 8) _ 1.25
- 9a) _ 1.00
- 9b) _ 1.00
- 10) _ 1.00
- 11a) _ 1.00
- 11b) _ 1.00
- 12a) _ 1.00
- 12b) _ 1.00
- 13a) _ 1.00
- 13b) _ 1.00
- 13c) _ 1.00

PROVA DE AVALIAÇÃO ESCRITA SEM CONSULTA – VERSÃO B SEGUNDA, 21 DE FEVEREIRO DE 2022
SEM CONSULTA DURAÇÃO: 50 MINUTOS (COMPLEMENTAR) / 100 MINUTOS (EXAME) COM TOLERÂNCIA INCLuíDA

NOME: _____ NÚMERO: _____

Atenção: Justifique todas as suas respostas na medida do necessário, sem escrever testamentos. Pense, organize as ideias e escreva. Pode apresentar exemplos e/ou grafismos para melhor elucidar as suas respostas. Considere $1\text{TiB}=2^{40}$ $1\text{GiB}=2^{30}$ Bytes, $1\text{MiB}=2^{20}$ Bytes e $1\text{kiB}=2^{10}$ Bytes. Nas questões de código, pode dividir a zona de resposta a meio para ter o dobro de linhas.

TEÓRICA / PARTE 1

- 1) Considerando a arquitetura ARMv7, indique 4 registos para dados e 4 registos para guardar endereços para dados.

- 2) Apresente um possível aspeto negativo da passagem de parâmetros por Stack.

- 3) Apresente os passos na invocação e regresso de uma sub rotina, nomeadamente indicando que registos e instruções estão envolvidos? Não incluir os aspetos específicos do método de passagem de parâmetros.

4) Quais as diferenças entre “Memory Mapped I/O” e “Port-mapped I/O” nas entradas/saídas?

TEÓRICA / PARTE 2

5) Indique como funcionam os diferentes níveis de memória cache de um processador.

6) Descreva a estrutura de um caractere (sequência de bits) na comunicação série.

- 7) Num sistema operativo real é possível criar programas em *assembly* e utilizar os recursos desse sistema, como por exemplo escrever uma mensagem na consola / terminal?

- 8) Que registos podem/devem ser utilizados para invocar uma chamada ao System Dispatcher / OS API, a partir do *assembly*?

PRÁTICA / ASSEMBLY

- 9) Responda às questões abaixo, tendo em conta o código assembly ARMv7 apresentado:

Código:

```

;R0 with a value
;R1 with STRING1 address, R2 with STRING2 address
START    cmp      r0, #0
          beq      STRING1
STRING2   mov      r0, r2
STRING1   mov      r0, r1
ENDCHECK
```

- a) De que tipo é estrutura de “decisão e controlo” utilizada no código?

b) Qual o “valor” do registo **R0** no final?

[illegible]

10) Tendo um vetor/string de valores de 8 bit, elabore um programa em assembly ARMv7 capaz de contar as maiúsculas (supondo código ASCII). A string tem formato tipo C (acaba com 0). ‘A’ e ‘Z’ representam os valores inteiros respetivos.

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There is no text or other markings on the paper.

11) Tendo por base o assembly ARMv7 e o emulador VisUAL responda às seguintes questões:

- a) Desenvolva uma sub-rotina que determine o número total de bits com o valor “1” de número de 32 bit, passado com parâmetro. Os registros de entrada e saída devem ser definidos por si.

[illegible]

- b) Desenvolva uma sub-rotina que determine o número total de bits com o valor “1” de um vetor de valores de 32 bits utilizando (invocando) a sub-rotina anteriormente definida.

[illegible]

PRÁTICA / ASSEMBLY / PLATAFORMA GNU

12) Considere o trecho de código seguinte (suponha que as equivalências / valores estão corretos):

Código:	
error:	<code>mov r7, #OS_WRITE</code>
	<code>mov r0, #OS_STDERR</code>
	<code>ldr r1, =errmsg</code>
	<code>mov r2, #31</code>
	<code>swi 0</code>

a) Indique qual o objetivo / o que faz o código.

b) O que representam os valores / registos **R0**, **R1**, **R2** e **R7**?

13) Novamente com base no assembly ARMv7 e um sistema com as ferramentas GNU:

a) Desenvolva uma sub-rotina que determine o somatório dos valores de um vetor de números de 8 bit. A interface com a sub-rotina deve ser a seguinte: **R0** (endereço do primeiro elemento do vetor) e **R1** (número de valores do vetor); retorno **R0** (somatório).

<div style="border-bottom: 1px solid black; margin-bottom: 5px;"></div> <div style="border-bottom: 1px solid black; margin-bottom: 5px;"></div> <div style="border-bottom: 1px solid black; margin-bottom: 5px;"></div> <div style="border-bottom: 1px solid black;"></div>	<div style="border-bottom: 1px solid black; margin-bottom: 5px;"></div> <div style="border-bottom: 1px solid black; margin-bottom: 5px;"></div> <div style="border-bottom: 1px solid black; margin-bottom: 5px;"></div> <div style="border-bottom: 1px solid black;"></div>
---	---

b) Indique que elementos / diretivas adicionais devem ser colocados no código, de forma a ser possível invocar a sub-rotina a partir de um programa em C.

[illegible]

c) Apresente um trecho de código / dados em C que utiliza a sub-rotina definida anteriormente.

[illegible]

Área extra para respostas (identificar qual a pergunta)

This image shows a single page of white paper with horizontal blue or grey ruling lines. The lines are evenly spaced and run across the width of the page, leaving small margins at the top and bottom. There are no vertical margin lines, and the page is completely blank except for the lines themselves.
