

Nome: _____ Nº. Mec. _____

Notas Importantes!

1. Verifique, para as questões de escolha múltipla, qual a resposta correcta e assinale com uma cruz a sua escolha na tabela ao lado. Nestas questões, por cada resposta incorrecta será descontada, à cotação global, 1/3 da cotação da respectiva pergunta.
2. Durante a realização do teste não é permitida a permanência na sala de calculadoras, telemóveis ou outros dispositivos electrónicos.

	a	b	c	d
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				

I

1. Um endereço de memória externa num sistema computacional é:
 - a. a gama de posições de memória que a CPU pode referenciar.
 - b. um número único que identifica cada posição de memória.
 - c. a informação armazenada em cada posição.
 - d. um índice de um registo de uso geral.
2. A arquitectura MIPS é caracterizada por:
 - a. possuir 32 registos de uso geral de 32 bits cada.
 - b. ser do tipo *load-store*.
 - c. possuir poucos formatos de instrução.
 - d. todas as anteriores.
3. Na arquitectura MIPS, os campos de uma instrução tipo "R" designam-se por:
 - a. "opcode", "rs", "rt" e "imm".
 - b. "opcode" e "address".
 - c. "opcode", "rs", "rt", "rd", "shamt" e "imm".
 - d. nenhuma das anteriores.
4. A instrução virtual "li \$t0, 0x10012345" da arquitectura MIPS decompõe-se na seguinte sequência de instruções nativas:
 - a. "lui \$1, 0x2345" seguida de "ori \$t0, \$1, 0x1001".
 - b. "ori \$t0, \$1, 0x1001" seguida de "lui \$1, 0x2345".
 - c. "lui \$1, 0x1001" seguida de "ori \$t0, \$1, 0x2345".
 - d. "ori \$t0, \$1, 0x2345" seguida de "lui \$1, 0x1001".
5. Nas instruções de acesso à memória da arquitectura MIPS é utilizado o modo de endereçamento:
 - a. indirecto por registo.
 - b. registo.
 - c. imediato.
 - d. directo.
6. O formato de instruções tipo "I" da arquitectura MIPS é usado nas instruções de:
 - a. salto condicional.
 - b. aritméticas em que somente um dos operandos está armazenado num registo.
 - c. acesso à memória de dados externa.
 - d. todas as anteriores.

7. Considerando que no endereço de memória acedido pela instrução "lb \$t0, 0xFF(\$t1)" está armazenado o valor 0x82, o valor armazenado no registo destino no final da execução dessa instrução é:
- a. 0xFF.
 - ☒ b. 0x82.
 - c. 0xFFFFFFFF82.
 - d. 0xFF82.
8. A instrução virtual "bgt \$t8,\$t9,target" da arquitectura MIPS decompõe-se na seguinte sequência de instruções nativas:
- a. "slt \$1,\$t8,\$t9" seguida de "bne \$1,\$0,target".
 - b. "slt \$1,\$t9,\$t8" seguida de "bne \$1,\$0,target".
 - ☒ c. "slt \$1,\$t8,\$t9" seguida de "beq \$1,\$0,target".
 - d. "slt \$1,\$t9,\$t8" seguida de "beq \$1,\$0,target".
9. Os endereços mínimo e máximo para os quais uma instrução "bne" presente no endereço 0x00430210 pode saltar são:
- ☒ a. 0x00000000, 0xFFFFFFFF.
 - b. 0x00000000, 0x0FFFFFFC.
 - c. 0x00428214, 0x00438213.
 - d. 0x00410214, 0x00450210.
10. A instrução "jal funct" executa sequencialmente as seguintes operações:
- ☒ a. $SPC = SPC + 4$, $\$ra = SPC$, $SPC = funct$.
 - b. $SPC = SPC + 4$, $SPC = funct$, $\$ra = SPC$.
 - c. $\$ra = SPC$, $SPC = funct$.
 - d. Nenhuma das anteriores.
11. Os endereços mínimo e máximo para os quais uma instrução "j" presente no endereço 0x00430210 pode saltar são:
- a. 0x00428214, 0x00438213.
 - b. 0x00000000, 0xFFFFFFFF. ☒
 - c. 0x00410214, 0x00450210.
 - d. 0x00000000, 0x0FFFFFFC.
12. Segundo a convenção de utilização de registos da arquitectura MIPS, uma subrotina não necessita de salvar os registos com os prefixos:
- ☒ a. \$a, \$v, \$s.
 - b. \$s, \$v, \$t.
 - c. \$a, \$v, \$t.
 - d. \$a, \$s, \$t.
13. Na arquitectura MIPS a *stack* é gerida de acordo com os seguintes princípios:
- a. cresce no sentido dos endereços mais altos, apontando o registo \$sp para a última posição ocupada.
 - ☒ b. cresce no sentido dos endereços mais baixos, apontando o registo \$sp para a última posição ocupada.
 - c. cresce no sentido dos endereços mais altos, apontando o registo \$sp para a primeira posição livre.
 - d. cresce no sentido dos endereços mais baixos, apontando o registo \$sp para a primeira posição livre.

14. Numa ALU, a detecção de *overflow* nas operações de adição algébrica é efectuada através:
- do “ou” exclusivo entre o *carry in* e o *carry out* da célula de 1 bit mais significativa.
 - da avaliação do bit mais significativo do resultado.
 - do “ou” exclusivo entre o bit mais significativo e o menos significativo do resultado.
 - do “ou” exclusivo entre os 2 bits mais significativos do resultado.
15. A decomposição numa sequência de adições e subtrações, de acordo com o algoritmo de Booth, da quantidade binária $101101_{(2)}$ é:
- $-2^0 + 2^1 - 2^2 + 2^4 - 2^5$
 - $2^0 - 2^1 + 2^2 - 2^4 + 2^5$
 - $2^0 - 2^1 + 2^2 + 2^3 - 2^4 + 2^5$
 - $-2^0 + 2^1 - 2^2 - 2^3 + 2^4 - 2^5$
16. A quantidade real binária $1011,11000000_2$ quando representada em decimal é igual a:
- 12,6
 - 11,75
 - 3008,0
 - 1504,0

II

Apresente o diagrama de blocos de um multiplicador de números de 16 bits sem sinal, iterativo e otimizado em termos da dimensão dos seus elementos funcionais. Indique os registos onde devem ser carregados os operandos e lido o resultado, a função de cada um dos restantes blocos, bem como as respectivas dimensões.

III

Considere as tabelas a seguir apresentadas. Admita que o valor presente no registo \$PC corresponde ao endereço da primeira instrução, que nesse instante o conteúdo dos registos é o indicado, e que vai iniciar-se o *instruction fetch* dessa instrução.

Endereço ...	Dados ...	Opcode	Funct	Operação		
0x10010040	0xFEC81248	0	0x20	add	\$4	0x00000F03
0x10010044	0x00410312	0	0x22	sub	\$5	0x10010050
0x10010048	0xC630F731	0	0x24	and	\$6	0x10010040
0x1001004C	0x3A509DB0	0	0x25	or	\$7	0x0040003C
0x10010050	0x8421C630	0x02		j	\$8	0x10010054
0x10010054	0x5FF38C29	0x04		beq	\$9	0x00000004
...	...	0x05		bne	SPC	0x0040002C
		0x08		addi		
		0x0C		andi		
		0x23		lw		
		0x2B		sw		

```

L1: beq $5,$6,L2
    lw  $7,0($5)
    and $8,$7,$4
    sw  $8,4($5)
    addi $5,$5,-8
    j   L1
L2:   ...

```

- Traduza para código máquina do MIPS o trecho de código correspondente às seis instruções da tabela da direita (expressando o resultado em hexadecimal) e indique o endereço de memória em que cada uma se encontra. Justifique todos os passos da sua resposta.
- Indique o conteúdo dos registos do CPU e das posições do segmento de dados apresentadas, após a execução do trecho de código fornecido, isto é, imediatamente antes de ser iniciado o *instruction fetch* da instrução armazenada no endereço correspondente à etiqueta L2. Justifique adequadamente a sua resposta.