

Nome: _____ Nº de estudante: _____

Atenção: Este teste tem 13 questões em 8 páginas, num total de 200 pontos.

Parte I — Questões de Escolha Múltipla

Cada questão tem uma resposta certa. Respostas erradas não descontam.

As respostas às questões de escolha múltipla devem ser assinaladas com × na grelha seguinte.

Apenas as respostas indicadas na grelha são consideradas para efeitos de avaliação.

| | Questão | | | | | | | | | |
|-------|---------|---|---|---|---|---|---|---|---|----|
| Opção | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| A | | | | | | | | | | |
| B | | | | | | | | | | |
| C | | | | | | | | | | |
| D | | | | | | | | | | |

Pontos: _____ / 100

- [10] 1. O código-máquina da instrução `addi $t1, $s3, -4` é:
- A. 0x22690004 B. 0x2269ffff C. 0x21330004 D. 0x2133ffff
- [10] 2. Assumindo que os valores iniciais dos registos `$t1` e `$t2` são respetivamente 0xABABABAB e 0xCDCDCDCD, qual é o valor do registo `$t1` após a execução do seguinte fragmento de código:
- ```

sll $t2, $t1, 14
srl $t1, $t2, 8
or $t1, $t1, $t2

```
- A. 0xEAEAEACO B. 0x00EAC000 C. 0xEAEACACO D. 0xEAEFCDCD
- [10] 3. Qual das seguintes afirmações sobre uma memória *cache* do tipo *write-through* é falsa?
- A. O conteúdo da memória principal está sempre atualizado.
- B. No caso de uma falta num acesso de leitura (*read miss*) o valor é lido da memória principal e colocado na *cache* atualizando a etiqueta e alterando o valor de *v* para 1.
- C. No caso de uma falta num acesso de escrita (*write miss*) o valor é escrito na memória principal e na memória *cache* atualizando a etiqueta e alterando o valor de *v* para 1.
- D. No caso de um acerto num acesso de leitura (*read hit*) o valor é lido da memória *cache*.
- [10] 4. Um CPU está equipado com uma memória *cache* unificada cuja taxa de faltas,  $t_f$ , é de 10%. Assumindo que em média 20% das instruções de um programa acedem a dados (i.e., são *load* ou *store*), qual das seguintes alternativas de *split cache* apresenta o mesmo desempenho?
- A. I-*cache* com  $t_f = 8\%$  e D-*cache* com  $t_f = 20\%$ .
- B. I-*cache* com  $t_f = 20\%$  e D-*cache* com  $t_f = 8\%$ .
- C. I-*cache* com  $t_f = 5\%$  e D-*cache* com  $t_f = 5\%$ .
- D. I-*cache* com  $t_f = 10\%$  e D-*cache* com  $t_f = 20\%$ .

- [10] 5. Devido a uma avaria, os sinais `MemRead` e `MemToReg` estão permanentemente a 1. Após executar `add $t1,$t2,$t3` resulta `$t1=4`. Este valor corresponde ao:

- A. conteúdo da memória na posição 4
- B. resultado de `$t2 + $t3`
- C. endereço de acesso à memória
- D. conteúdo da memória na posição dada por `$t2 + $t3`

- [10] 6. Qual é o valor de `$t1` após a execução da instrução assinalada com um asterisco (\*)?

```

 li $s0, 5
 li $a0, 10
 jal rotina
 and $v0, $v0 $zero
 add $t1, $v0, $s0 # (*)
 ...
rotina: add $v0, $a0, $a0
 addi $ra, $ra, 4
 jr $ra

```

- A. 0      B. 25      C. 15      D. 5

- [10] 7. Para um dado programa, o processador P1 com  $F_1 = 1$  GHz apresenta o mesmo tempo de execução que o processador P2 com  $F_2 = 1,25$  GHz. O tempo de execução de P1 fica maior que o de P2 se:

- A. se passar a usar  $F_2 = 1$  GHz
- B. se aumentar o valor do CPI médio de P2
- C. se reduzir o valor do CPI médio de P1
- D. se aumentar 1,3 vezes o período do relógio de P1

- [10] 8. O tempo de execução de um programa está repartido entre a execução de instruções da classe A (60 % do tempo) e da classe B (40 % do tempo).

Qual das seguintes alterações leva ao melhor desempenho?

- A. diminuir para metade o tempo de execução das instruções de classe A;
- B. diminuir o tempo de execução das instruções de classe B para um quarto do tempo original;
- C. reduzir o tempo de execução das instruções de classe A para um terço e aumentar o tempo de execução das instruções de classe B para o dobro;
- D. reduzir 1,5 vezes o tempo de execução das instruções de classe A e reduzir o tempo de execução das instruções de classe B para metade.

- [10] 9. Um programa gasta 75 % do tempo em transferências de dados para outro computador via rede sem fios. Quantas vezes é preciso aumentar a velocidade de transferência para obter uma redução do tempo de execução do programa (*speedup*) de duas vezes?

- A. 4      B. 1,5      C. 3      D. 2

- [10] 10. Considerar o seguinte fragmento de código *assembly*.

```

 slt $t2, $t1, $t0
 bne $t2, $zero, L1

```

Para que valores de `$t0` e `$t1` é que o salto condicional é tomado?

- A. `$t0 = -10` e `$t1 = -10`
- B. `$t0 = -10` e `$t1 = 0`
- C. `$t0 = -10` e `$t1 = -50`
- D. `$t0 = 10` e `$t1 = 50`

Nome: \_\_\_\_\_ Nº de estudante: \_\_\_\_\_

**Parte II — Questões de Resposta Aberta****Atenção:** Responder diretamente no enunciado. **Justificar** todas as respostas.

11. O programa abaixo aplica a sub-rotina `calc` aos elementos de uma sequência `nums`, acumula os resultados das invocações e apresenta o total acumulado antes de terminar a execução.

[15] (a) Completar o programa.

```

.data
nums: .word 0x00000005, 0xFFFF8C00, 0xA00800C
nelem: .word 3

init: la $s0, nums
 _____ $s1, nelem # nº de elementos a processar
 addi $s2, $zero, 0
ciclo: beq $s1, _____, L1 # terminar ciclo
 lw $a0, 0($s0)
 _____ calc # invocar sub-rotina
 add $s2, $s2, _____ # usar o resultado
 addi $s0, $s0, 4
 addi $s1, $s1, _____
 _____ ciclo
L1: move $a0, _____
 li $v0, 1
 syscall # escrever valor inteiro no monitor
 li $v0, 10
 syscall # terminar execução

Sub-rotina calc tem 1 argumento
calc: xor $v0, $v0, _____ # inicializar $v0 com zero
LC1: beq $a0, $zero, LC2 # terminar?
 andi $t0, $a0, 1
 add $v0, $v0, $t0
 srl $a0, $a0, 1
 j LC1
LC2: jr _____ # fim da subrotina

```

- [10] (b) Determinar o número de instruções executadas pela sub-rotina `calc` quando é chamada pela primeira vez.
- Assumindo que as instruções de salto (condicional ou incondicional) têm  $CPI=2$  e todas as outras têm  $CPI=1$ , determinar também o valor de  $CPI$  médio. Indicar todos os cálculos.

- [10] (c) Considerar apenas argumentos de valor  $2^k$  ( $k$  inteiro,  $0 \leq k \leq 31$ ). Explicar o resultado da sub-rotina e determinar o número de instruções executadas em função de  $k$ .

Prova continua na próxima folha

Nome: \_\_\_\_\_ Nº de estudante: \_\_\_\_\_

12. A tabela seguinte apresenta o conteúdo (em hexadecimal) de uma memória *cache* do tipo *write-back* com 8 blocos de 8 bytes usada como *D-cache* num CPU com endereços de 16 bits.

| bloco | conteúdo |    |    |    |    |    |    |    | etiqueta | v | d |
|-------|----------|----|----|----|----|----|----|----|----------|---|---|
|       | 7        | 6  | 5  | 4  | 3  | 2  | 1  | 0  |          |   |   |
| 0     | aa       | cc | de | hf | 34 | 33 | 11 | 01 | 235      | 1 | 0 |
| 1     | bb       | ad | 45 | 4f | af | de | 21 | 99 | 391      | 1 | 1 |
| 2     | cc       | 34 | ab | 1f | 56 | cd | ff | ff | 023      | 1 | 1 |
| 3     | dd       | 67 | 22 | 2b | 32 | 56 | 32 | 21 | 198      | 0 | 1 |
| 4     | ee       | 32 | 11 | 9f | aa | ba | ab | bb | 311      | 1 | 0 |
| 5     | ff       | 10 | 00 | 04 | 01 | 02 | 03 | 04 | 278      | 0 | 0 |
| 6     | 11       | 03 | 41 | 32 | cc | dd | ee | ff | 212      | 1 | 1 |
| 7     | 22       | 01 | 65 | 01 | 05 | 06 | 07 | 08 | 387      | 0 | 1 |

- [5] (a) Como é decomposto o endereço para acesso à memória *cache*? Justifique.

- [15] (b) Indique (se possível) o valor (byte) em memória principal no endereço 0xc467. Justifique.

- [15] (c) Explique quais as alterações que ocorrem na *cache* e na memória principal durante a leitura do valor (byte) residente no endereço `0xe48d`.

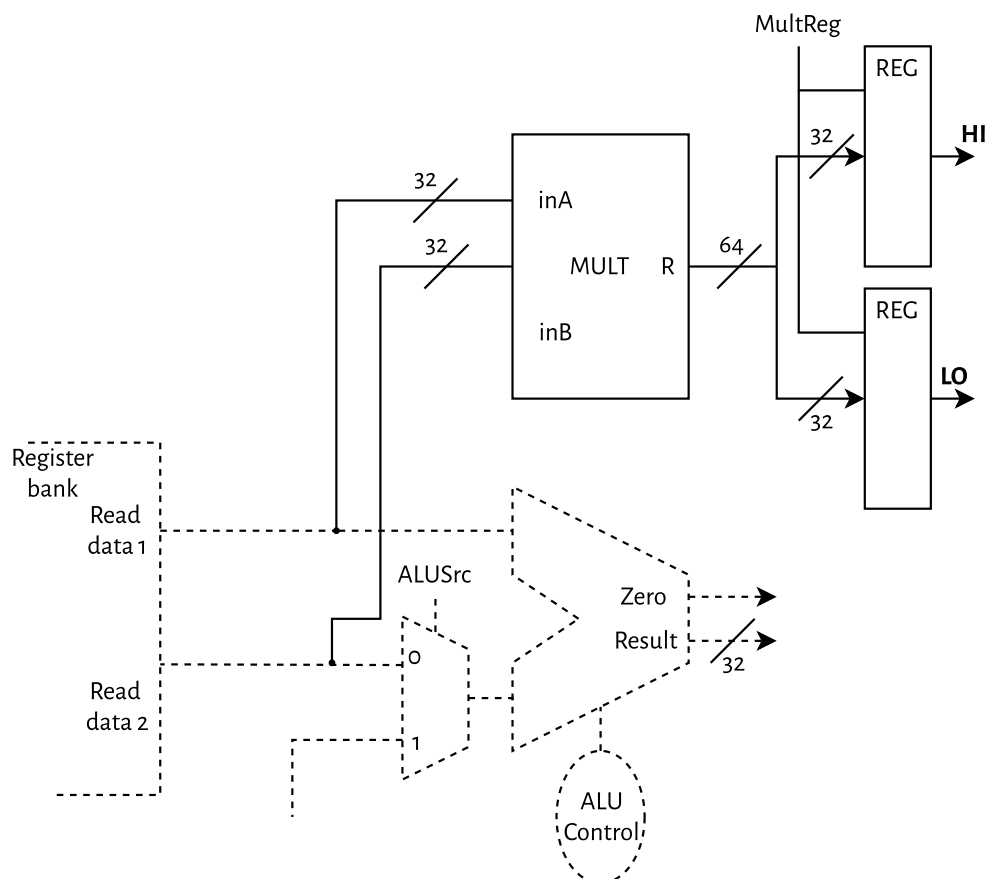
Prova continua na próxima folha

Nome: \_\_\_\_\_ Nº de estudante: \_\_\_\_\_

13. Considerar a organização interna do CPU MIPS simplificado estudada nas aulas.

- [10] (a) Para que instrução (ou instruções) os sinais de controlo assumem simultaneamente os seguintes valores:  $\text{RegWrite}=1$  e  $\text{ALUSrc}=1$ ? Justifique.

- (b) Para implementar a instrução `mult`, **acrescentaram-se** ao CPU os componentes indicados a traço contínuo na figura abaixo. O módulo `MULT` é um multiplicador com operandos de 32 bits e resultado de 64 bits. O sinal `MultReg` habilita a escrita nos registos `HI` e `LO` e é gerado pelo controlador `ALU Control`. O sinal de relógio e os restantes módulos do CPU não estão representados.



- [10] i. Indique, justificando, o valor de todos os sinais de controlo do CPU durante a execução de `mult`. Indique também o valor de `MultReg` durante a execução das restantes instruções. Ter em consideração a forma como a instrução é codificada.

- [10] ii. Pretende-se acrescentar agora suporte para as instruções `mfhi` e `mflo`. Explique, com o auxílio de um diagrama, como é que isso pode ser feito e quais os valores de todos os sinais de controlo do CPU. Quaisquer sinais de controlo adicionais devem ser produzidos pelo controlador `ALU Control`.

Fim do enunciado.