

ARQUITETURA DE COMPUTADORES
Prof. Calebe Conceição

Prova 1

Ao responder a prova, observe com atenção o que dizem os enunciados das questões. Evite rasuras, especialmente nas questões cuja resposta implica em marcação única na folha. Use o verso da folha para responder as questões dissertativas/argumentativas. Desejo uma excelente prova. Sucesso!

Questão 1) (2 pontos) O que é o modelo de Von Neumann (0,3 pontos), liste quais são os ciclos para execução de um programa (1,2 pontos), e comente sobre a sua importância para a organização dos computadores modernos, especialmente no que tange à flexibilidade (0,5 pontos).

90 **Questão 2)** (0,5 pontos) Linguagens de descrição de hardware (HDL) como VHDL e Verilog são usadas para capturar o funcionamento do hardware a ser desenvolvido. Ambas as linguagens possuem diferentes estilos de projeto, representando níveis de abstração distintos com que podemos especificar o funcionamento do hardware com mais ou menos detalhes, deixando menos ou mais trabalho para o software de síntese. Sobre isso, especificamente sobre Verilog, avalie as assertivas que seguem.

- I. Ao usar a modelagem comportamental, o projetista deixa para o sintetizador a responsabilidade de decidir sobre quais as funções lógicas que serão usadas para implementar o circuito.
- II. Ao usar a modelagem dataflow, o projetista define o comportamento como um conjunto de operações lógicas que serão usadas na elaboração da netlist.
- III. Na modelagem estrutural, cabe ao projetista definir as instâncias específicas de componentes, como também a quantidade de flip-flops e as conexões necessárias.

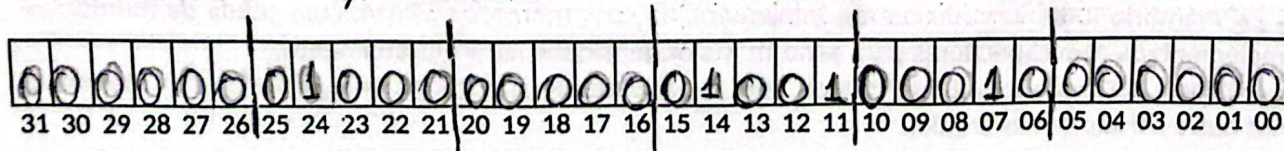
Quais estão corretas?

- a) Apenas os itens I e II estão certos.
b) Apenas os itens I e III estão certos.
c) Apenas os itens II e III estão certos.
d) Apenas um item está certo.
e) Todos os itens estão certos.

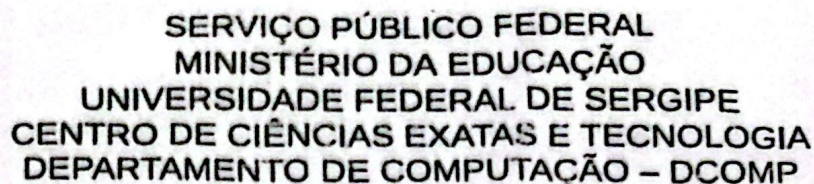
Para responder as questões que seguem, você pode consultar o material no final da prova.

Questão 3) (1 ponto) Escolha uma das instruções MIPS indicadas a seguir, marcando-a com um X. Então apresente no quadro abaixo o código binário que a representa. Escreva com clareza e evite rasuras. Destaque os campos estendendo ligeiramente a linha vertical, conforme expresso no quadro. Observe.

- a) `addi $a2, $a1, -1` ☒ b) `sll $t1, $t0, 2` c) `lw $t4, 0($t1)` d) `beq $t1, $t3, 27`

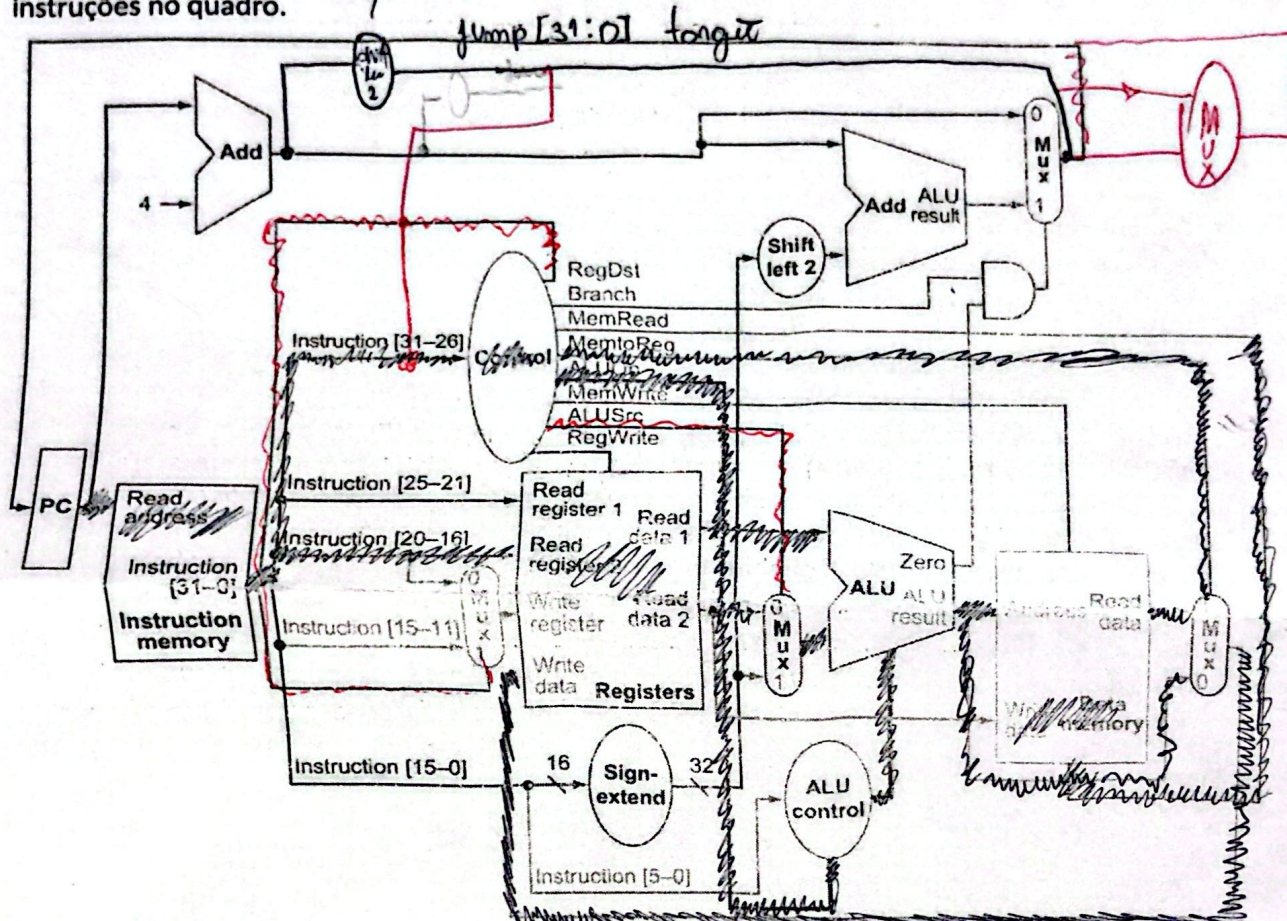


Questão 4) (1 ponto) O que são pseudo-instruções? Cite um exemplo e justifique. *Woue*



Página 2/5

Questão 5) (1,5 pontos) Observe o esquemático a seguir. Para a instrução escolhida na questão anterior, pinte o caminho de dados que é ativo na sua execução, destacando fios e blocos funcionais. Observe as instruções no quadro.



013 Questão 6) (1 ponto) O que falta no esquemático anterior para dar suporte à instrução Jump? Comente e desenhe, complementando o esquemático acima. Seja preciso e evite rasuras.

15 Questão 7) (1,5 pontos) Avalie as assertivas que seguem sobre memória, julgando-as em Verdadeiro (V) ou Falso (F). Anote de forma legível as letras V ou F entre os parênteses.

- a. (V) Em uma memória volátil, se a energia for desligada, a informação armazenada é perdida. Assim, é necessária a alimentação elétrica constante para manter o conteúdo da memória. Em uma memória não volátil, o conteúdo é estável e não exige uma fonte de alimentação constante.
- b. (F) A memória RAM é normalmente implementada com memórias SRAM, cuja célula de memória é implementada com capacitores para garantir maior densidade no armazenamento.
- c. (F) HDDs e SSDs são memórias voláteis com capacidade de armazenamento maior que as memórias dos outros níveis da hierarquia.
- d. (F) A memória cache L1 (Level 1) é um exemplo de memória DRAM (Dynamic Random Access Memory), que é uma tecnologia de memória volátil e não demanda refresh.

c. (F) Considerando a pirâmide que representa a hierarquia de memória, à medida que subimos da base em direção ao topo obtemos: menor latência, menor capacidade de armazenamento, e menor custo por byte.

10 Questão 8) (1,5 pontos) O que faz o código assembly MIPS ao lado?

0,5 a) Descreva de forma direta e sucinta (1,0 pontos).

0,5 b) Inclua comentários que indiquem o que fazem as linhas iniciais, e cada um dos labels (main, lb, enc, nenc, exit) (0,5 pontos).

```

1 .data # declara uma variável do tipo word chamada vetor com 10
2 vetor: .word 5, 10, 15, 20, 25, 30, 35, 40, 45, 50
3
4 .text
5 .globl main
6
7 main: # função principal, pede um inteiro user e declara valores de variáveis que
      # serão usados no código
8   li $v0, 5 # lê um int
9   syscall
10  move $t0, $v0 # guarda int em $t0
11
12  la $t1, vetor # aponta vetor em $t1
13  li $t2, 0 # $t2 recebe 0
14  li $t3, 10 # $t3 recebe 10
15  # função lb busca o número lido pelo usuário no vetor, se não
16  lb: # não ache nada para nenc e caso ache só para nenc.
17  bge $t2, $t3, nenc # se $t2 >= $t3, volta nenc
18  lw $t4, 0($t1) # lê a palavra 0 do vetor em $t1, e coloca em $t4
19  beq $t4, $t0, enc # se $t4 == $t0, volta enc
20  addi $t1, $t1, 4 # $t1 = $t1 + 4
21  addi $t2, $t2, 1 # $t2 = $t2 + 1 == $t2++
22  j lb # jump para lb
23
24 enc: # função enc, encontra o número impresso e guarda de user user
25  li $v0, 1 # imprime inteiro para o user, que também é a
26  move $a0, $t2 # $a0 = $t2 posição no vetor. volta para
27  syscall # imprime $t2
28  j exit
29  # fim da função
30 nenc: # função nenc, não encontra o número ($t0 != vetor), retorna
31  li $v0, 1 # "-1" impresso no tela do user. e finaliza
32  li $a0, -1 # de user exit
33  syscall # imprime -1
34  j exit
35
36 exit: # se user chamado essa função encerra o programa
37  li $v0, 10
38  syscall

```

#BoaProva