

Exame 2020 - AC1

1. Considere uma arquitetura em que o ISA, memória wa (word=16 bits).
espaço de endereçamento do processador 30 bits; qual a máxima
memória em bits?
2. $\$f_0 = 0x\text{FF}80\ 0000 \rightarrow$ qual a representação em notação científica?
- $-1,0 \times 2^{128}$
 - NAN
 - $-0,00000000 \cdot 2^{-127}$
 - infinito
3. O formato das instruções tipo "I" do MIPS é usado nas instruções.
- aritméticas em que ambos os operandos estão armazenados em registros
 - deslocamento em que imm identifica o n.º de deslocamentos a efetuar
 - salto incondicional
 - acesso à memória de dados sistema.
4. MIPS r' do tipo "load-store", significa que:
- os operandos das operações aritméticas e lógicas podem ser usados imediatamente antes das
 - as inst. de load store apenas podem ser usadas imediatamente antes das op. aritméticas e lógicas
 - nesta arquitetura foi dada especial importância à implementação das inst. lu e su de forma a não comprometer o desempenho.
 - os operandos das op. aritméticas e lógicas apenas podem ser usados em registros internos.
5. Nas instruções de acesso à memória do MIPS é utilizado o modo de endereçamento:
- tipo registro
 - imediato
 - indireto por registro
 - indireto por registro com offset
6. Considerando que no end. memória acessado por lb \$t0, 0x00FF(\$t1)
está o valor 0x82, o valor armazenado no registro destino é:
- 0xFFFFF82
 - 0xFF82
 - 0x82
 - 0xFF

- 7) A instrução jal label ~~executa~~ ^{executa} sequencialmente as seguintes operações:
- PC = PC + 4; PC = label; \$ra = PC
 - \$ra = PC; PC = label; PC = PC + 4
 - \$ra = PC; PC = PC + 4; PC = label
 - PC = PC + 4; \$ra = PC; PC = label

8) tipo Harvard é caracterizada por:

- permite o acesso à int. e dados no mesmo ciclo de relógio.
- segmentos de memória independentes para dados e para endereços.
- partilhar a mesma memória entre dados e baseamentos de endereços.
- 2 baseamentos de dados e um baseamento de endereços.

9) A representação normalizada e coordenada para o π próximo de acordo IEEE 754, precisão simples do π = 100, 110110000000000000010110₂

- 1,000 (...) $\cdot 2^{-2}$
- 1,001101100 $\cdot 2^{-2}$
- 1, (...) 101 $\cdot 2^{-2}$
- 1, (...) 110 $\cdot 2^{-2}$

10) O resultado mult \$t0, \$t4 é representável em 32 bits se:

- HI for $\neq 0$
- HI = 0xFFFFFFFF
- HI = 0x00000000
- HI = for uma extensão do bit + significativo de LO

11) instr virtual la \$t0, label

- ori \$t0, \$t0, 0x001 lui \$t1, 0x1001
- ori \$t0, \$t0, 0x0001 lui \$t1, 0x0040
- lui \$t1, 0x0040 ori \$t0, \$t1, 0x0001
- lui \$t1, 0x1001 ori \$t0, \$t1, 0x0001

12) O trecho de código que permite atribuir o valor 0xFF à variável "i" indiretamente através do ponteiro "p" é:

- a.

```
int i;  
int *p;  
p = *i;  
*p = 0xFF
```
- b.

```
int i;  
int *p = 0xFF;  
p = &i;  
i = p;
```
- c.

```
int i;  
int *p;  
p = &i;  
*p = 0xFF
```
- d.

```
int i;  
int *p;  
i = &p;  
*i = 0xFF
```

13) detecção de ~~overflow~~ overflow numa op. de adição de n-int com sinal faz-se através.

- a. xor ^{teste} a bits + significativos do resultado.
- b. avaliação do carry out do bit + significativo do resultado.
- c. xor entre o carry in e carry out da célula do 1 bit + significativo.
- d. avaliação do bit + sign. do resultado.

14) implementações ~~de~~ single-cycle do Mips:

- a. existe uma única ALU ~~que~~ para realizar todas as operações aritméticas e lógicas numéricas para executar num único ciclo de relógio qualquer uma das instruções suportadas.
- b. existem numéris específicas para código e dados para possibilitar o acesso a ambos os tipos de informação num único ciclo de relógio.
- c. Todas as operações de leitura e escrita são síncronas com o sinal de relógio.
- d. existem registos à saída dos elementos operativos fundamentais para guardar valores a utilizar no ciclo de relógio seguinte.

15) A unidade de controlo de uma implementação multi-cycle:

- a. é um elemento combinatório que gera os sinais de controlo em função do campo funct do código máquina da instrução.
- b. é um elemento combinatório que gera os sinais de controlo em função do campo opcode do código máquina da inst.
- c. é uma máquina de estados com um n° de estados igual ao n° de fases da inst. + longa.
- d. é uma máquina de estados em que o 1° e o 2° estados são comuns à execução de todas as instruções.

16) multicycle; Na segunda e terceira fase de execução de uma inst. do salto condicional (" bq/bne "), a ALU é usada, pela ordem indicada para.

- | | | | | | | |
|-------------|------|-----------|--------------------|------|-----------|----------------------|
| a. calcular | PC+4 | e | comparar | os | registros | (operandos da inst.) |
| b. " | " | e | " | " | " | " |
| c. comparar | os | registros | (op. da instrução) | e | calcular | o valor do BTA |
| d. calcular | o | valor | de | PC+4 | e | valor do BTA |

17) A frequência de relógio de uma implementação pipeline da arquitetura Mips.

- a. é limitada pelo menor dos tempos de atraso dos elementos operativos Memória, ALU e File Register
- b. é definida por forma a emitir stalls e delay slots
- c. é limitada pelo maior dos atrasos dos elementos operativos envolvidos na execução da instrução + longa.
- d. é limitada pelo maior dos tempos de atraso dos elementos operativos Memória, ALU e File Register.

18) A técnica de ~~for~~ forwarding/bypassing num processador Mips pipeline permite.

- a. escrever o resultado de uma instrução no File Register antes da escrita chegar à etapa WB.
- b. trocar a ordem de execução das instruções de forma a resolver hazards de dados.
- c. utilizar como operando de uma instrução, um resultado produzido por outra instrução que se encontra numa etapa + avançada do pipeline.
- d. " " " " " " etapa + recuada do pipeline.

19) Numa implementação single-cycle da arquitetura Mips, a freq. máxima da operação imposta pela instn. de leitura da memória de dados é, assumindo os atrasos a seguir indicados.

20) Um hazard de controle numa implementação pipelined de um processador ocorre quando

- a. a unidade de controle desconhece o opcode da instrução que se executa na etapa ID.
- b. um dado acesso de hardware é necessário para realizar, no mesmo ciclo de relógio, 2 ou mais operações relativas a instruções que se encontram em diferentes etapas do pipeline.
- c. existe uma dependência entre o resultado calculado por uma instrução e o operando usado por ... mais atrás do pipeline.
- d. é necessário fazer o instruction fetch de ... instruções e existe, numa etapa mais ~~avanzada~~ avançada de pipeline, ... instruções que ... terminou e que pode alterar o fluxo de execução.

21) instrução virtual bgt \$t8, \$t9, target → instrução nativa

- a. slt \$t1, \$t9, \$t8 ... bne \$t1, \$0, target
- b. slt \$t1, \$t8, \$t9 ... beq \$t1, \$0, target
- c. slt \$t1, \$t9, \$t8 ... beq \$t1, \$0, target
- d. slt \$t1, \$t8, \$t9 ... bne \$t1, \$0, target

22) admita que se pretende inicializar o conteúdo do registro \$f4 com quantidade real 2.0. A sequência de instn. é:

a.

li.s. \$f4, 2
cvt.s.w \$f4, \$f0

b.

li \$t2, 2
mtc1 \$t2, \$f4

d.

li \$t0, 2
mtc1 \$t0, \$f0
mov.s \$f4, \$f0

c.

li \$t0, 0x4000
mtc1 \$t0, \$f4

23) código máquina inst. SW \$3, -128(\$4), representado em hexadecimal, (opcode = 0x2B)

- a. 0xAC648080
b. 0xAC838080
c. 0xAC64FF80
d. 0xAC83FF80

24) considerando que $f_2 = 0 \times 3A60\ 0000$ e $f_4 = 0 \times BA60\ 0000$,
resultado do sub.s f_0, f_2, f_4

- a. $\$f_0 = 0 \times 3 \text{ AE } 0000$
b. $\$f_0 = 0 \times 8000 \text{ } 0000$
c. $\$f_0 = 0 \times 3 \text{ ? } \text{E} 0000$
d. $\$f_0 = 0 \times \text{ } 00000000$

25) $a = 0x00000000$, representa uma quantidade codificada em hexadecimal IEEE 754 simples "a" e em notação decimal.

- $-3, 25 \cdot 2^1$
- $-16, 25 \cdot 2^1$
- $-0, 16 \cdot 25 \cdot 2^{-1}$
- $6, 25 \cdot 2^{-2}$

26) Considere as seguintes frequências relativas de instruções de um programa a executar no Mips:

Operação	Frequência Relativa
lw	20%
sw	10%
R	50%
beq/bne	15%
j	5%

Assumindo um multicycle a operar a 100 MHz

A melhoria de desempenho proporcionada pelo multicycle a opera a ~~100~~ 100 MHz relativamente a single cycle a opera 20 MHz é de:

- a. 5
b. 1,25
c. 0,8
d. 1

27)

28) } Exercícios pipeline
29) }

30) Os processadores da hp. KObid implementa um total de 59 instruções. Todas as instruções são codificadas em 32 bits, num formato com 5 campos "opcode", 3 campos para registos internos e um campo para valores imediatos na gama $[-4096, 4095]$.

- a. Se o número de registos internos é 16 e o opcode tem 7 bits.
- b. " " " 32 " " 6 bits.
- c. " " " 16 " " 6 ".
- d. " " " 8 " " 8 ".