Docente:	Prof.	Dr.	Clodoaldo	<b>A</b> ]	M	Lima.
Disconto						

Discerne.	·	NO. USP.	•

No LICD.

1ª Questão) Relaciona a coluna da esquerda com a coluna da direita

(I) Multicore	(	)	Múltiplos pipelines que operam em paralelo
(II) Companying alima	1	`	E

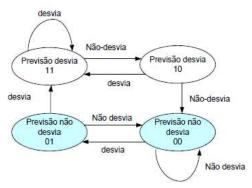
(II) Superpipeline ( ) Execução de instruções fora de ordem em um pipeline.

(III) Superescalar ( ) Pipelines com grande número de estágios.

(IV) Pipeline dinâmico ( ) Múltiplos processadores compartilhando um espaço de endereços.

(V) Multiprocessadores ( ) Múltiplos processadores em um único encapsulamento

- 2ª Questão) Usando o sistema de previsão local de desvio de 2 bits, mostrado na Figura1, um certo loop é executado duas vezes:
- a) considerando-se que o estado inicial seja 00, calcular a porcentagem de acertos e erros de previsão, considerando-se que o loop termina com 10 iterações;
- b) comparar com o caso em que não use esse sistema de previsão, e apenas considere que a previsão seja sempre de desvio;
- c) comparar com o sistema de previsão de um bit



3ª Questão) Identificar as situações de dependência (WAW, WAR, RAW) na seguinte sequência de código, do MIPS64:

DIVD F1, F3, F5 ADDD F4, F1, F9 SD F4, 0(R1) SUBD F1, F10, F14 MULD F9, F10, F8

4ª Questão) Assinale verdadeiro (V) ou falso (F).

(Lembre-se: um item assinalado incorretamente anula um item assinalo corretamente)

- ( ) RISC apresenta poucos formatos de instrução e muitos registradores de uso genérico, enquanto CISC possui instruções de vários comprimentos (no mesmo conjunto)
- ( ) Arquitetura superpipeline baseia-se no aumento das unidades funcionais de forma que seja possível executar mais de uma instrução em cada ciclo de relógio
- ( ) Na RISC a complexidade esta no compilador, enquanto na CISC a complexidade esta no microprograma

- ( ) Uma arquitetura super-escalar consiste em aumentar o número de estágios da pipeline, conseguindo diminuir Tcc e aumentar a frequência de relógio
  ( ) Arquitetura vetorial especifica uma série de operações a realizar em vários dados, numa só instrução
  ( ) Uma arquitetura com grau de grau de super-escalaridade igual a 2 apresenta 2 ciclos de penalização (5 instruções) nos saltos previstos incorretamente
  ( ) Programas compilados para arquitetura CISC possuem garantia que serão menores que os compilados para RISC.
  ( ) No mecanimo de write back uma escrita modifica o dado na cache e memória juntos
  ( ) No caso em que não há escalonamento dinâmico, as instruções são emitidas pela ordem com que são geradas pelo compilador, executadas pela mesma ordem e terminadas ainda em ordem
  ( ) Tamanhos e posições das instruções são fixos e alinhados de acordo com o tamanho de uma palavra
- 5ª Questão) Considere o conjunto de instruções abaixo

em RISC

					Latência
I1	div	F6	F6	F4	4
I2	lw	F2	45(R3)		1
I3	mult	F0	F2	F4	3
I4	div	F8	F6	F2	4
I5	sub	F10	F0	F6	1
<b>I</b> 6	add	F6	F8	F2	1

- a) Identifique as situações de dependência (WAW, WAR, RAW) na seguinte sequência de código acima, do MIPS64:
- b) Apresente uma sequência de termino em ordem e outra em fora de ordem (que execute no menor tempo)
- 6ª Questão) Considere o trecho de programa no quadro abaixo e os conteúdos iniciais de registradores e posições de memória relevantes. Convenções: X bolha, F flush do pipeline, -- para estágio não usado, -\_ adiantamento ou leitura após escrita no mesmo ciclo. Estágios do pipeline: BI(Busca), DI (Decodificação), EX (Execução) MEM (Memória) WB (Writeback)

addi \$t4, \$zero, 2	Conteúdos iniciais da memória e dos registradores relevantes:
root : add \$t1, \$t2, \$t3	\$t1=0x100, \$t2=0x100, \$t3=0x100, \$t4=0x100
lw \$t3, 0x100(\$t1)	Mem [0x100-0x103] = 0x002345AB
sw \$t3, 0x200(\$t1)	Mem [0x200-0x203] = 0x00000000A
subi \$t4, \$t4, 2	Mem [0x300-0x303] = 0x000000000
beq \$t4, \$t3, root	Mem [0x400-0x403] = 0x00CD5F00
addi \$t3, \$t3, 0x100	

- a) Simule a execução completa do programa (considere unidade de adiantamento).
- b) O que a unidade de adiantamento (forward) está fazendo durante o quinto ciclo de execução? Se algumas comparações estiverem sendo feitas, mencione-as.
- 7ª Questão) Considere a seguinte sequencia de instruções, e assuma que estas sejam executadas em um pipeline com 5 estágios (BI(Busca), DI (Decodificação), EX (Execução) MEM (Memória) WB (Write-back))

Sequencia Instruções
Iw \$1, 40 (\$6) add \$2, \$3, \$1 sw \$2, 20(\$4) add \$1, \$6, \$4 and \$1, \$1, \$4

- a) Quais dependências são conflitos (hazards) que podem ser resolvidos com adiantamento? Quais dependências que são conflitos e irão provocar a parada (bolhas) na execução?
- b) Se não há adiantamento ou detecção de conflito, insira nops para assegura a execução correta e desenhe o diagrama de execução do pipeline para este código
- c) Repita o item anterior, mas adicione nops somente quando um conflito n\u00e3o pode ser evitado por mudando ou rearranjando estas instru\u00e7\u00e3es. Voc\u00e9 pode assumir o registrador R7 para guardar valores tempor\u00earios em seu c\u00e9digo modificado.
- d) Um conflito estrutural (duas instruções tentando acessar a memória) pode ser resolvido pelo compilador inserindo uma instrução nops?
- e) Suponha as instruções abaixo. Qual o procedimento a ser adotado pela unidade de detecção de conflito load \$1,(10) \$2 add \$2, \$1, \$3
- f) Apresente o teste de conflito realizado no estagio EX e MEM pela unidade de adiantamento.
- 8ª. Questão) Mostrar o resultado (décimo ciclo) do uso do placar(scoreboard) para a sequência de instruções, considerando-se que a instrução LD leva 1 ciclo para execução; MULD, 6 ciclos. ADDD e SUBD levam 3 ciclos; e DIVD, 20 ciclos.

LD F2, 34(R2) LD F6, 45(R3) MULD FO, F2, F4 SUBD ,F8, F6, F2 DIVD F10, F0, F6 ADDD F6, F8, F2

Estado da il	nstruç	ão			X Es	creve				
Instrução	j	k	Emite	Lê E. Oper fin			,			
	¥									
M 100 100	E Table	L					J			
Estado das	FU									
Tempo restante	FU	Busy	Ор	dest Fi	S1 Fj	S2 Fk	FU Qj	FU	Fj? Rj	Fk?
	Int Mult1 Mult2 Ad Div									
Registro de	estado	dos	resul	tados						
		F0	F2	F4	F6	F8	F10	F12		F30
	FU									

9ª Questão) Mostrar o resultado do uso do algoritmo de Tomasulo, com os mesmos números de ciclos para a mesma sequência de instruções da questão anterior.

