Universidade de São Paulo
Escola de Artes, Ciências e Humanidades
Docente: Prof. Dr. Clodoaldo A M Lima.
Discente: _____ No. USP:_____

1ª Questão) (1,5 ponto) Identificar as situações de dependência (WAW, WAR, RAW) na seguinte sequência de código, do MIPS64:

I1 - DIVD F1, F3, F5
I2 - ADDD F4, F1, F9
I3 - SD F4, 0(R1)
I4 - SUBD F1, F10, F14
I5- MULD F9, F10, F8

WAW – I4-I1 - 0,5
WAR – I4- I2; I5- I2 - 0,5
RAW- I1- I2; I3- I2 - 0,5

2ª Questão) (1,5 ponto) Assinale verdadeiro (V) ou falso (F).
(Lembre-se: um item assinalado incorretamente anula um item assinalo corretamente)

(F) CISC apresenta poucos formatos de instrução e muitos registradores de uso genérico, enquanto RISC possui instruções de vários comprimentos (no mesmo conjunto)
(F) Arquitetura superpipeline baseia-se no aumento das unidades funcionais de forma que seja possível executar mais de uma instrução em cada ciclo de relógio
(V) Arquitetura multicore consiste múltiplos processadores em um único encapsulamento
(F) Na CISC a complexidade está no compilador, enquanto na RISC a complexidade está no microprograma
(F) Uma arquitetura super-escalar consiste em aumentar o número de estágios da pipeline, conseguindo diminuir Tcc e aumentar a frequência de relógio
(V) Arquitetura vetorial especifica uma série de operações a realizar em vários dados, numa só instrução
(V) Uma arquitetura com grau de grau de super-escalaridade igual a 2 apresenta 2 ciclos de penalização (5 instruções) nos saltos previstos incorretamente
(F) Programas compilados para arquitetura CISC possuem garantia que serão menores que os compilados para RISC.
(F) No mecanismo de write back uma escrita modifica o dado na cache e na memória juntos
(V) 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
(V) Tamanhos e posições das instruções são fixos e alinhados de acordo com o tamanho de uma palavra em RISC
(V) No escalonamento dinâmico consiste em realizar a execução de instruções fora de ordem em um pipeline
(F) O escalonamento estático não bloqueia a execução de instruções em estágios posteriores do pipeline, nem bloqueia a emissão de novas instruções
(F) No escalonamento estático independentemente da ordem na qual as instruções chegam ao datapath, esta pode executá-las por uma ordem diferente desde que não quebre o fluxo de dados

3ª Questão) (2,0 pontos) Considere o conjunto de instruções abaixo

					Latência
I1	div	F1	F6	F4	3
I2	lw	F2	45(R3)		2
I3	mult	F4	F2	F4	3
I4	div	F8	F1	F2	4
I5	sub	F10	F4	F1	1
I6	add	F1	F8	F10	1

a) (1,1 ponto) Identifique as situações de dependência (WAW, WAR, RAW) na seguinte sequência de código acima, do MIPS64.

WAW I6-I1,

WAR I3-I1, I6-I4, I6-I5

RAW **I3-I2**, **I4-I1**, **I4-I2**, **I5-I3**, **I5-I1**, **I6-I4**, **I6-I5**

b) (0,4 ponto) Apresente uma sequência de termino em ordem e outra em fora de ordem (que execute no menor tempo)

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14
I1	I2		I1	I2	I3	I4		I3	I5	I4	I5	I6	I6

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14
I1	I2		I1, I2	I3	I4		I3	I5	I4 I5	I6	I6		

4ª. Questão) (2,5 ponto) Mostrar o resultado (décimo ciclo) do uso do placar(scoreboard) para a sequência de instruções, considerando-se que a instrução LD leva 2 ciclo para execução; MULD, 4 ciclos. ADDD e SUBD levam 3 ciclos; e DIVD leva 20 ciclos.

LD F2, 34(R2)
ADD F2, F2, R4
LD F6, 45(R3)
MULD FO, F2, F4
SUBD ,F8, F6, F2
DIVD F10, F0, F6
ADDD F6, F8, F2

Estado da instrução

Instrução	j	k	Emite	Lê Oper	EX final	Escreve resultado
LD F2 34(R2)			1	2	4	5
ADD F2 F2 R4			6	7	10	
LD F6 45(R3)			7	8	10	
MULT F0, F2, F4			8			

1,0 cada linha 0,25

Estado das FU

Tempo restante	FU	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU Qj	FU Qk	Fj? Rj	Fk? Rk
	Int	Y	LD	F6		R3				Y
	Mult1	Y	MULTD	F0	F2	F4	ADD		N	Y
	Mult2									
	Ad	Y	ADD	F2	F2	R4			Y	Y
	Div									

0,5 — 0,15 / 0,2 / 0,15

Registro de estado dos resultados

	F0	F2	F4	F6	F8	F10	F12	...	F30
FU	Mult1	Ad		Int					

0,5 — cada item errado anula item certo

0,2 0,15 0,15

5ª Questão) (2,5 ponto) Mostrar o resultado do uso do algoritmo de Tomasulo, com os mesmos números de ciclos para a mesma sequência de instruções da questão anterior.

Estado da instrução

0,5 0,25 0,25

Instrução	j	k	Emite	EX final	Escreve resultado
LD F2, 34(R2)			1	4	5
ADD F2, F2, R4			2	8	9
LD F6 45(R3)			3	6	7
MultD F0, F2 F4			4		
SUBD F8, F6, F2			5		
Div F10, F0, F6			6		
ADD F6, F8 F2			7		

	Busy	Ender (A)
LD1	N	
LD2	N	
LD3	N	

Estações de reserva

Tempo restante	RS	Busy	Op	S1 Vj	S2 Vk	RS Qj	RS Qk
	Add1	N	ADD		F2	ADD3	
	Add2	N					
	Add3	N	SUBD	M(A2)	M(A1)+R(F4)		
	Mult1	N	MultD	M(A1)+R(F4)	F4		
	Mult2	N	DivD		M(A2)	MultD	

0,5

Registro de estado dos resultados

	F0	F2	F4	F6	F8	F10	F12	...	F30
FU	MultD	M(A1)+R(F4)		ADD1	ADD3	Mult2			

0,5

6ª Questão) (1,5 ponto) Suponhamos que melhoramos uma máquina fazendo todas as instruções de ponto-flutuante serem executadas 5 vezes mais rápido.

a) (0,75 ponto) Se o tempo de execução de certo benchmark antes do melhoramento é de 10s, qual seria o speedup se metade dos 10s é despendida em instruções de ponto-flutuante?

$T_{old} = 10$

$$T_{new} = T_{old} * \left[(1 - fraction) + \frac{fraction}{speed\_up}\right]$$

$$T_{new} = T_{old} * \left[(1 - 0.5) + \frac{0.5}{5}\right]$$

$$T_{new} = T_{old} * [0.5 + 0.1]$$

$$T_{new} = T_{old} * 0.6$$

$$Speed_{up} = \frac{T_{old}}{T_{new}} = \frac{1}{0.6} = 1.66$$

Se colocar 1,66 % ao invés de 166% - descontar - 0.05

Calculo tempo, mas não cálculo o speedup - desconta 0.35

b) (0,75 ponto) Estamos procurando um benchmark para testar a nova unidade de ponto-flutuante acima, e queremos que o benchmark todo mostre um speedup de 3. Um benchmark é executado em 100s, com o antigo hardware de ponto-flutuante. Quanto do tempo de execução as instruções de ponto-flutuante devem corresponder nesse programa para que possamos produzir o speedup desejado nesse benchmark?

$$T_{new} = T_{old} * \left[(1 - fraction) + \frac{fraction}{speed\_up}\right]$$

$$1 = \frac{T_{old}}{T_{new}} * \left[(1 - fraction) + \frac{fraction}{speed\_up}\right]$$

$$1 = 3 * \left[(1 - fraction) + \frac{fraction}{5}\right]$$

$$1 = 3 * \left[\frac{5 - 4 * fraction}{5}\right]$$

$$5 = 15 - 12 * fraction$$

$$fraction = \frac{10}{12} = 0.83$$

Resultado correto, mas cálculou tempo e não proporção - desconta 0.25

Resultado incorreto, mas equações corretas desconta 0,