

UFPR - Depto de Informática
CI086 - Tópicos em Arquitetura de Computadores

Primeira Prova

1) Mostre como o trecho de código abaixo seria escalonado dinamicamente para execução num processador que usa o algoritmo de Tomasulo. O processador é capaz de emitir até três instruções por ciclo, sendo uma de inteiro (**add**, **sub**, **mult**, **bra**), uma de ponto flutuante (**add.d**, **mult.d**), e uma de memória (**ld**, **st** de inteiros ou PF). A janela de observação é infinita. As latências são definidas na tabela abaixo. Considere duas voltas do laço. Quantos ciclos são necessários para completar (aposentar) a última instrução da segunda volta? A resposta deve ser entregue na folha em anexo. [15 pontos]

<pre> loop: ld.d f0,0(r4) ld.d f2,0(r6) mult.d f4,f2,f0 add.d f0,f4,f8 # f8 escalar st.d f4,0(r8) st.d f0,8(r8) addi r4,r4,8 addi r6,r6,8 addi r8,r8,8 bne r8,r9,loop # r9 limite </pre>	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;"></td> <td style="width: 15%; border-bottom: 1px solid black;">produz</td> <td style="width: 15%; border-bottom: 1px solid black;">usa</td> <td style="width: 15%; border-bottom: 1px solid black;">ciclos</td> </tr> <tr> <td style="border-bottom: 1px solid black;">mult PF</td> <td style="border-bottom: 1px solid black;">alu PF</td> <td style="border-bottom: 1px solid black;">3</td> <td></td> </tr> <tr> <td style="border-bottom: 1px solid black;">adic PF</td> <td style="border-bottom: 1px solid black;">alu PF</td> <td style="border-bottom: 1px solid black;">2</td> <td></td> </tr> <tr> <td style="border-bottom: 1px solid black;">alu PF</td> <td style="border-bottom: 1px solid black;">sto PF</td> <td style="border-bottom: 1px solid black;">2</td> <td></td> </tr> <tr> <td style="border-bottom: 1px solid black;">ld</td> <td style="border-bottom: 1px solid black;">sto</td> <td style="border-bottom: 1px solid black;">4</td> <td></td> </tr> <tr> <td style="border-bottom: 1px solid black;">ld</td> <td style="border-bottom: 1px solid black;">qquer</td> <td style="border-bottom: 1px solid black;">2</td> <td></td> </tr> <tr> <td style="border-bottom: 1px solid black;">inteiro</td> <td style="border-bottom: 1px solid black;">qquer</td> <td style="border-bottom: 1px solid black;">0</td> <td></td> </tr> <tr> <td style="border-bottom: 1px solid black;">bra</td> <td style="border-bottom: 1px solid black;">qquer</td> <td style="border-bottom: 1px solid black;">0</td> <td></td> </tr> </table>		produz	usa	ciclos	mult PF	alu PF	3		adic PF	alu PF	2		alu PF	sto PF	2		ld	sto	4		ld	qquer	2		inteiro	qquer	0		bra	qquer	0	
	produz	usa	ciclos																														
mult PF	alu PF	3																															
adic PF	alu PF	2																															
alu PF	sto PF	2																															
ld	sto	4																															
ld	qquer	2																															
inteiro	qquer	0																															
bra	qquer	0																															

núm iter	instrução	número do ciclo em que				comentário
		emite	exec	mem	wr CDB	

- 2) O que são *excessões precisas*? Por que são desejáveis? [5 pontos]
- 3) Desenhe um diagrama do processador com *pipeline* simples de cinco estágios e indique no diagrama os estágios onde podem ocorrer excessões, ou o estágio mais próximo da busca onde pode-se detectar a excessão. Indique todas as excessões que podem ocorrer. [5 pontos]
- 4) Projete o algoritmo do controlador de uma cache primária com escrita preguiçosa (*write-back*). Os blocos são de 8 palavras e a cache entrega ao processador a palavra crítica primeiro. O algoritmo deve controlar as duas interfaces da cache: L1-CPU e L1-L2. Especifique as entradas, saídas e estruturas de dados. [10 pontos]
- 5) Enuncie a Lei de Amdahl. Como esta lei influencia no projeto de hierarquias de memória? [5 pontos]

Segunda Prova

- 1) Quais as diferenças, com relação ao desempenho global do sistema, entre a sincronização com uma *barreira* com N processos em (i) um multiprocessador com memória fisicamente compartilhada (através de um barramento), e (ii) um multiprocessador com memória logicamente compartilhada mas fisicamente distribuída pelos nós de uma rede? [8 pontos]
- 2) Como você alteraria o código de uma aplicação para evitar falso compartilhamento? O que poderia ser feito pelo compilador, e o que necessitaria de diretivas pelo programador? [5 pontos]
- 3) Quais as diferenças, em termos de desempenho global, dos protocolos de coerência de caches por invalidação e por atualização? [5 pontos]
- 4) Escreva a equação do tempo de acesso para leitura de um setor num disco e explique sucintamente a contribuição de cada termo para o desempenho global do sistema de E/S. [5 pontos]

5) A rede do CM-5 usa roteamento da minhoca (*wormhole routing*). Compare a eficiência de roteamento por re-despacho (*store-and-forward*) com o roteamento da minhoca para uma máquina com 256 nós, através da árvore quaternária, usando mensagens de 20 bytes (4+16). Cada comutador tem latência de $0,25\mu\text{s}$ e a taxa de transferência é de 20MB/s. [5 pontos]

$$\text{RE-DESPACHO } L_{rd} = \#c \times (\mathcal{T}_c + \mathcal{T}_t) \quad \mathcal{T}_t = |m|/b$$

6) Desenhe um diagrama mostrando a atualização de um bloco num sistema de discos com RAID-5 com largura 5 (os blocos podem ser gravados paralelamente em até 5 unidades de disco). [6 pontos]

7) Considere um disco com 16 setores e duas trilhas. Desenhe um diagrama mostrando o leiaute dos setores nas trilhas, com os setores dispostos de forma a se obter o máximo de desempenho. Justifique sua resposta. [6 pontos]

Exame Final

1) O trecho de código abaixo calcula a expressão de ponto flutuante $E = A \times B + C \times D$, e os endereços de A,B,C,D,E estão armazenados em r1,r2,r3,r4,r5 respectivamente. [40 pontos]

a) Calcule o número de ciclos necessários para executar este trecho de código (número de ciclos entre a emissão do primeiro LD até a emissão do último ST, inclusive) numa CPU com pipeline simples em ordem que não tem adiantamento. O circuito de dados inclui uma unidade de acesso à memória (LD/ST), um somador de ponto flutuante e um multiplicador de PF. Suponha que todos os LDs tem uma latência de 2 ciclos, multiplicação tem 4 ciclos e adição 2 ciclos. Escrever o resultado em registradores de PF demora 1 ciclo. Suponha também que todas as unidades funcionais são segmentadas e ignore conflitos na escrita de resultados.

b) Simule a execução do código original num processador com pipeline simples com renomeação de registradores e emissão fora-de-ordem. Ignore riscos estruturais que não sejam a emissão de uma instrução por ciclo. Mostre como o código é executado e calcule o número de ciclos. Se necessário, acrescente à tabela ao lado uma função de renomeação de registradores.

1 ld.d	f0, 0(r1)	ciclo	instr decodificada entra emissão	instr emitida entra execução	ciclo escrita
2 ld.d	f1, 0(r2)				
3 mul.d	f0, f0, f1				
4 ld.d	f2, 0(r3)				
5 ld.d	f3, 0(r4)				
6 mul.d	f2, f2, f3				
7 add.d	f0, f0, f2				
8 st.d	f0, 0(r5)				

2) Uma CPU superescalar emite 4 instruções por ciclo, e tem relógio de 2 GHz (0.5ns/ciclo). Uma falta na L1 custa, no mínimo, 10 ciclos. Quais os números mínimo e máximo de instruções que poderiam ser executadas durante uma falta na L1? Justifique. [10 pontos]

3) Considere as políticas de escrita preguiçosa e forçada, ainda alocação e não-alocação de um bloco numa falta na escrita. (i) Especifique cuidadosamente as quatro combinações possíveis (EP+aloc, EP+nãoAloc, EF+aloc, EF+nãoAloc). (ii) Qual destas é a melhor para memória virtual? Justifique. [20 pontos]

4) Considere um barramento de memória que interliga a cache secundária à memória. Os blocos da cache tem 4 palavras de largura e o clock do barramento é de 200MHz (5ns/ciclo). Os CIs de memória tem tempo de acesso de 47ns. O barramento pode ter, no máximo, 64 vias para dados (mas pode ter tantas quanto necessário para endereço e controle).

(i) Defina e descreva o projeto de um barramento e da interface do circuito de memória que, simultaneamente, minimize a latência e maximize a vazão. [15 pontos]

(ii) Calcule a vazão sustentada e a latência do seu projeto da questão 1.1, considerando três casos, de transferências em rajadas com 1, 2 ou 4 palavras. [15 pontos]