
Arquitetura e Organização de Computadores

Prof. Tiago Gonçalves Botelho

Lista de exercícios 8 – Referente a aula – Computadores com um conjunto reduzido de instruções

- 1 – Explique por que o uso de instruções RISC favorece o uso de pipelines.
- 2 – O que é o gap semântico? Quais problemas podem ocorrer provenientes dele?
- 3 – Instruções CISC tem pontos positivos? Quais?
- 4 – Analisando a tabela de verificação do número de instruções de máquina e referências à memória causadas por um tipo de comando em linguagem de alto nível, quais as duas chamadas dispõem de mais instruções ponderada? Por quê?
- 5 – Por que é difícil projetar uma arquitetura de instruções assembly próximo de linguagens de alto nível?
- 6 – Qual problema poderia ocorrer caso algumas instruções fossem buscadas antecipadamente em um pipeline?
- 7 – Explique como funciona a janela de registradores.
- 8 – Explique as duas opções para armazenamento de variáveis globais em registradores.
- 9 – Cite e explique duas comparações entre memória cache e registradores.
- 10 – O que diferencia um registrador simbólico, ou virtual de um real?
- 11 – Faça o grafo de interferência para a sequência de uso de registradores, considerando três registradores reais:

A = 1 a 5	E = 5 a 12
B = 2 a 8	F = 0 a 2
C = 0 a 3	G = 3 a 6
D = 6 a 10	H = 8 a 11
- 12 – Como um compilador otimizador pode melhorar o desempenho de uma arquitetura RISC?
- 13 – Por que uma arquitetura RISC está sujeita a um maior número de interrupções? O que acontece na arquitetura CISC com relação a essa característica?
- 14 – Simule os códigos em assembly utilizando: (I) estrutura sequencial, (II) pipeline de dois caminhos e (III) pipeline de três caminhos. Para cada um existe a possibilidade de utilizar desvio atrasado otimizado?

CÓDIGO (A):

```
lw $t0,$s0
lw $t1,$s1
addi $t2,$t0,8
sub $t2,$t1,$t0
j label
add $t1, $t1, $t0
mul $t1, $t0, $t1
label:
sub $t3, $t2, $t1
sw $t3,$s2
```

CÓDIGO (B):

```
li $t2,0
loop:
    lw $t3,0($t0)
    bne $t3,$zero, cont
    j exit
cont:
    add $t2, $t2, $t3
    addi $t0,$t0,4
j loop
exit:
```