



Disciplina: Arquitetura e Organização de Computadores II (6895-Turma 1)

Trabalho

Objetivo: O trabalho aqui proposto tem como objetivo avaliar conceitos de processadores que fazem escalonamento dinâmico de instruções.

Instruções

- O trabalho poderá ser feito em grupo de até 3 integrantes.
- O trabalho proposto poderá ser implementado em uma das seguintes linguagens:
 - C/C++;
 - Java;
 - Python;

Obs: caso a dupla deseje utilizar outra linguagem deverá comunicar o professor.

- O trabalho vale de 0,0 a 10,0 e corresponde a 3ª avaliação periódica.
- O trabalho deverá ser entregue via classroom até o dia 08/12/2021.
- O nome do arquivo enviado deve seguir o padrão:
Aluno1_RAXXXXXX_Aluno2_RAXXXXXX_Aluno3_RAXXXXXX .zip;
 - O formato para submissão deve ser .zip, .rar, ou .tar;

Descrição: Deve-se implementar um simulador simplificado para uma máquina que escala dinamicamente as instruções (algoritmo de Tomasulo). As descrições para o hardware são as seguintes:

- Não é necessária a implementação de operações com ponto flutuante;
- A máquina deve ter memória de dados e instruções separadas;
 - Assuma que cada instrução cabe em uma posição da memória de instruções e cada posição da memória armazena um elemento inteiro
- O tamanho da memória de dados fica a critério da equipe, porém deve possuir no mínimo 512 posições;
- Assuma que o programa que está em execução pode ser armazenado inteiro na memória de instruções;
- A máquina deve ter 32 registradores de uso geral;
 - Os registradores devem ser nomeados de r0 até r31;
 - O registrador r0 deve sempre armazenar o valor constante 0;
 - Será utilizado para fazer as operações de atribuição junto a alguma operação aritmética de registrador com imediato, por exemplo:
 - movi r3,5 deverá ser escrito como addi r3, r0, 5;
- As unidades funcionais devem ser divididas da seguinte maneira:
 - Três unidades de memória para cálculos de endereço de *load* e *store*;
 - Três unidades funcionais para soma e subtração de inteiros;
 - As operações lógicas e de desvios devem ser feitas nas mesmas unidades funcionais que soma e subtração de inteiros;
 - Três unidades funcionais para multiplicação e divisão de inteiros;
- A quantidade de estações de reserva total da máquina deve ser de 48;
 - Dezesesseis para soma e subtração;
 - Dezesesseis para multiplicação e divisão
 - Dezesesseis para *loads* e *stores*;



Disciplina: Arquitetura e Organização de Computadores II (6895-Turma 1)

- As instruções, seus significados e a quantidade de ciclos, no estágio de execução, que cada uma delas usa é dado pela tabela 1:

Tabela 1: Conjunto de instruções

Tipo de Instrução	Representação da instrução	Significado	Quantidade de ciclos
Aritmética	add rd, rs, rt	Atribui à rd a soma de rs e rt $rd \leftarrow rs + rt$	5
	addi rd, rs, imm	Atribui à rd a subtração entre rs e um valor imediato $rd \leftarrow rs + imm$	5
	sub rd, rs, rt	Atribui à rd a soma de rs e rt $rd \leftarrow rs - rt$	5
	subi rd, rs, imm	Atribui à rd a subtração entre rs e um valor imediato $rd \leftarrow rs - imm$	5
	mul rd, rs, rt	Atribui à rd o produto entre rs e rt $rd \leftarrow rs * rt$	15
	div rd, rs, rt	Atribui à rd o quociente da divisão de rs por rt $rd \leftarrow rs \div rt$	25
Lógica	and rd, rs, rt	Atribuiu à rd o resultado da operação and bit a bit entre rs e rt $rd \leftarrow rs \& rt$	5
	or rd, rs, rt	Atribuiu à rd o resultado da operação or bit a bit entre rs e rt $rd \leftarrow rs rt$	5
	not rd, rs	Atribuiu à rd o resultado da operação not bit a bit $rd \leftarrow \sim rt$	5
Desvio	blt rs, rt, imm	Salta caso rs seja maior que rt Se $rs < rt$ então $pc \leftarrow imm$	5
	bgt rs, rt, imm	Salta caso rs seja menor que rt Se $rs > rt$ então $pc \leftarrow imm$	5
	beq rs, rt, imm	Salta caso rs e rt sejam iguais Se $rs = rt$ então $pc \leftarrow imm$	5
	bne rs, rt, imm	Salta caso rs e rt sejam diferentes Se $rs \neq rt$ então $pc \leftarrow imm$	5
	j imm	Salto incondicional $pc \leftarrow imm$	5
Memória	lw rd, imm(rs)	Carrega da memória para o registrador rd $rd \leftarrow M[imm + Rs]$	5
	sw rs, imm(rt)	Armazena o valor de rs na memória $M[imm + rs] \leftarrow rs$	5

Obs: nas instruções imm é uma constante inteira.

- As etapas de **busca e despacho** de instrução devem levar um ciclo cada;



Disciplina: Arquitetura e Organização de Computadores II (6895-Turma 1)

- Lembre-se que nos exemplos feitos em sala o ciclo de busca foi desconsiderado;
- Ao buscar uma instrução, ela deve ser colocada na fila de despacho.
 - O tamanho da fila de despacho poder ser determinado pela equipe.
- Uma instrução não pode realizar mais de uma etapa no mesmo ciclo, ou seja, se ela foi buscada no ciclo 1, ela não pode ser despachada nesse mesmo ciclo;
- A máquina **não** possui especulação, então deve-se tomar cuidado com desvios e operações de acesso à memória de dados;
- O arquivo de entrada que se deve receber é um arquivo contendo uma sequência de instruções. Assuma que a instrução que está na primeira linha será a primeira a ser executada, a que está na segunda linha será a segunda, e assim por diante;
- A cada ciclo deverá ser exibido:
 - O estado das estações de reserva;
 - O estado da memória de dados;
 - O estado dos registradores;

Problemas com Trabalhos COPIADOS:

Quem copiar terá o trabalho anulado (zerado), seja de outra dupla ou da internet.
Quem fornecer a cópia também terá o trabalho anulado (zerado).