

Resumo do trabalho final de Arquitetura e organização de computadores

Savio Mendes, Yago P. Koki, Pedro Bocalon

¹Instituto de Ciência e Tecnologia– Universidade Federal de São Paulo (UNIFESP)



Professora Denise Stringhini.

Este trabalho foi elaborado como parte das atividades finais da disciplina de Arquitetura e Organização de Computadores (AOC), buscando integrar os conceitos teóricos com uma abordagem prática através da implementação de um algoritmo em Assembly MIPS. O código desenvolvido tem como objetivo permitir ao usuário inserir uma lista de valores, ordená-los e calcular estatísticas fundamentais, como média, moda e mediana.

Introdução

A disciplina de AOC envolve a compreensão detalhada dos componentes de hardware e sua interação por meio de instruções em baixo nível. A linguagem Assembly MIPS é uma excelente ferramenta para exercitar esses conceitos, pois permite o controle direto sobre registradores, memória e execução de instruções básicas do processador.

Este projeto tem como objetivo implementar um programa em Assembly MIPS que solicita ao usuário uma lista de valores, armazena-os em memória, ordena os elementos e calcula estatísticas básicas. O programa também explora conceitos fundamentais da disciplina, como hierarquia de memória, organização de barramentos e execução de ciclos de clock.

O relatório é estruturado da seguinte maneira: a seção de Fundamentação Teórica apresenta os conceitos necessários para compreender o funcionamento do código. Em Métodos, descrevemos o funcionamento do algoritmo. A seção de Resultados Esperados analisa os benefícios educacionais da implementação, e a Conclusão reflete sobre os aprendizados e possíveis melhorias.

Fundamentação Teórica

A implementação do programa em Assembly MIPS abrange diversos conceitos fundamentais da disciplina:

Hierarquia de Memória: O programa armazena valores em regiões contínuas da memória, utilizando o espaço reservado na seção `.data`.

Ciclos de Clock: A execução das instruções é feita respeitando a ordem dos ciclos de clock do processador, otimizando a sequência de operações.

Barramentos: A troca de dados entre registradores e memória é realizada por meio de instruções `load (lw)` e `store (sw)`, representando o fluxo de informações pelo barramento.

Pipeline: A organização das instruções explora o conceito de pipeline, evitando stalls ao ordenar e calcular estatísticas.

Execução de operações matemáticas: O programa utiliza operações de soma, divisão e comparação para calcular a média, moda e mediana dos valores inseridos.

Materiais e Métodos

Materiais:

Ambiente de desenvolvimento MIPS, como o simulador MARS.

Conjunto de instruções Assembly MIPS.

Estruturas de memória para armazenar os valores e estatísticas.

Métodos:

Entrada de Dados: O usuário insere até 15 valores, armazenados em um vetor na memória.

Ordenação dos Valores: O algoritmo classifica os valores utilizando comparações e movimentação de dados na memória.

Cálculo de Estatísticas:

- Média: Calculada somando os valores e dividindo pelo total.
- Moda: Determinada pela contagem da frequência dos elementos.
- Mediana: Extraída do valor central do vetor ordenado.

Saída de Dados: Os resultados são exibidos ao usuário na tela.

Resultados Esperados

A implementação deste código em Assembly MIPS permite aos alunos consolidar o conhecimento sobre organização de computadores, reforçando a compreensão da hierarquia de memória, do funcionamento de registradores e da execução de código de baixo nível. Espera-se que a execução do programa facilite a visualização prática dos

conceitos abordados em sala de aula, aumentando o entendimento sobre a interação entre hardware e software.

Conclusão

O trabalho apresentado demonstrou como a implementação de um programa em Assembly MIPS pode ser utilizada para reforçar conceitos teóricos de Arquitetura e Organização de Computadores. A manipulação direta de registradores e memória proporcionou um aprendizado prático e aprofundado sobre a estrutura interna do processador e a gestão de dados em nível de hardware.

A abordagem adotada mostrou-se eficaz para ilustrar o funcionamento de instruções básicas do conjunto MIPS, permitindo a exploração de conceitos como ordenação, contagem de frequência e cálculo de estatísticas. Como trabalhos futuros, sugerimos a expansão do programa para incluir novos métodos de ordenação e a implementação de algoritmos otimizados para cálculo estatístico em Assembly.

Referências Bibliográficas

STALLINGS, William. *Arquitetura e Organização de Computadores*. 10. ed. São Paulo: Pearson, 2016.

TANENBAUM, Andrew S.; BOS, Herbert. *Arquitetura de Computadores*. 6. ed. São Paulo: Pearson, 2016.

WERBACH, Kevin; HUNTER, Dan. *Gamification*. 1. ed. Pennsylvania: Wharton School Press, 2012.

.