

1º Trabalho da Disciplina ELC1011

O 1º trabalho da disciplina ELC1011 – Organização de Computadores, poderá ser realizado individualmente ou em duplas. Quando desenvolvido em duplas, espera-se, que ambos os alunos desenvolvam o trabalho. A entrega do trabalho será realizada por meio da ferramenta Moodle, na data acordada com o professor da disciplina.

Enviar um arquivo comprimido (tipo ZIP) contendo o relatório do trabalho (em PDF) e os arquivos fontes dos programas desenvolvidos. O relatório do trabalho deverá conter as seguintes seções: introdução, objetivos, revisão bibliográfica, metodologia, experimento, resultados, discussão e conclusões e perspectivas. O código fonte deve estar completo e comentado.

No primeiro trabalho do curso será desenvolvido um programa para simular um subconjunto de instruções do processador MIPS. O programa será escrito em linguagem *assembly* para o processador MIPS. Uma ilustração do primeiro projeto da disciplina ELC1011 é apresentado na figura 1.

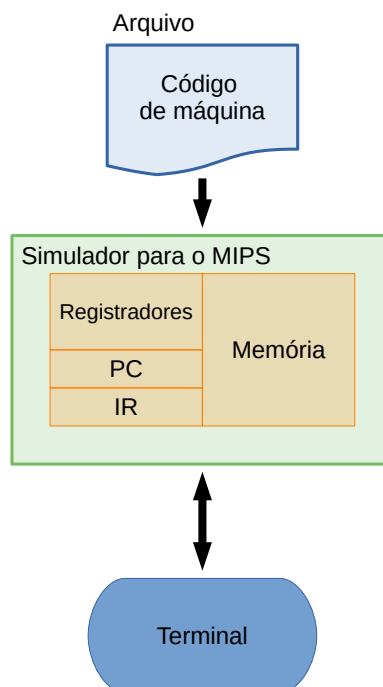


Figura 1: Ilustração do primeiro projeto da disciplina ELC1011.

As instruções em linguagem de máquina que serão executadas pelo programa simulador e os dados estáticos estão armazenadas em dois arquivos binários.

O simulador recebe por meio de um terminal os comandos do usuário. Os comandos serão:

lt <nome do arquivo> Faz a leitura do arquivo binário <nome do arquivo> com as instruções em linguagem de máquina. Ex.: l text.bin. Este comando faz a leitura das instruções em linguagem de

máquina do arquivo `text.bin`.

ld <nome do arquivo> Faz a leitura do arquivo binário <nome do arquivo> com os dados estáticos do programa. Ex.: `ld data.bin`. Este comando faz a leitura dos dados estáticos do arquivo `data.bin`.

r <numero de instruções> Realiza a simulação das instruções em linguagem de máquina. O número de instruções que serão simuladas é dada por <número de instruções>. Ex.: `r 10`. Este comando realiza a simulação de 10 instruções em linguagem de máquina, do endereço dado por PC.

d Apresenta no terminal o conteúdo atual dos registradores. Ex. `d`.

m <endereço inicial> <número de endereços> Este comando apresenta no terminal o conteúdo da memória. O número de endereços apresentado é dado por <número de endereços>, começando pelo endereço <endereço inicial>. Ex.: `m 0x10010000 10`. Este comando apresenta o conteúdo de 10 endereços, começando pelo endereço `0x10010000`.

Para a execução de uma instrução, o programa irá passar por 3 estados, como mostra a figura 2.

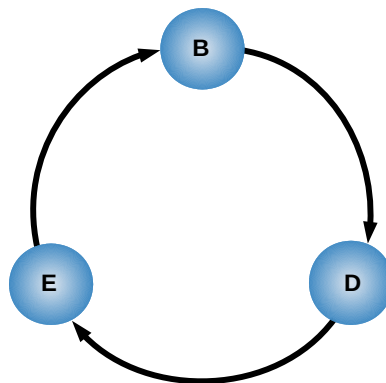


Figura 2: Estados simulados do processador: busca da instrução **B**, decodificação da instrução **D** e execução da instrução **E**.

No estado **B**, busca da instrução, o programa irá buscar a instrução do endereço dado pelo registrador PC (program counter). O próximo estado é a decodificação da instrução, estado **D**. Neste estado, a instrução é separada em seus campos e verificada a operação que deve ser realizada pelo simulador do processador MIPS. Neste estado também incrementamos o registrador PC. No estado **E**, execução da instrução, a instrução decodificada é executada. Na sequência, voltamos ao estado **B** e o ciclo recomeça para a próxima instrução. Para verificarmos a execução da instrução, apresentamos no terminal os registradores e, ou, os valores da memória, usando os comandos `d` ou `m`.

Será fornecido pelo professor o arquivo binário e as instruções em assembly equivalentes. As instruções que serão desenvolvidas no simulador, a programação do simulador e o cronograma de atividades serão discutidas em sala de aula.