

### Instruções e Observações:

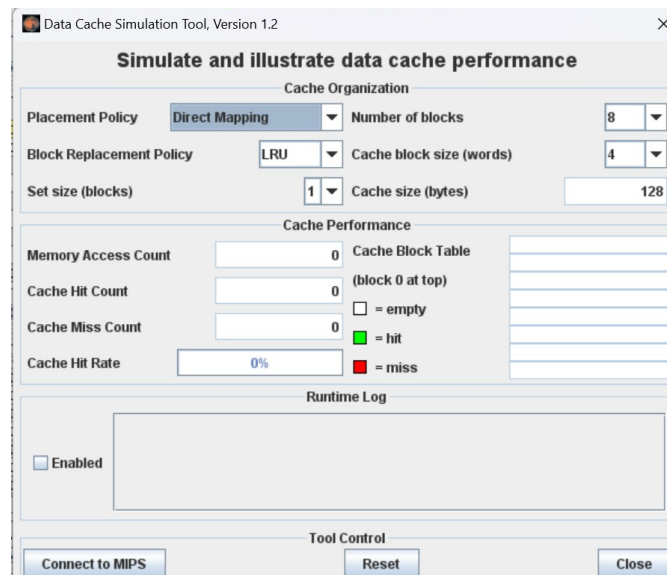
- 1 – Você deverá postar no Moodle os arquivos .doc (um para cada item solicitado abaixo), Informando os resultados obtidos.
- 2 – Tire um print de tela mostrando que os resultados das operações estão corretos. Você pode colar estes *print* de tela no arquivo .doc, inclua seu nome e número de matrícula e gere um pdf deste documento final. Este documento em pdf também deverá ser anexado no Moodle.

**Importante:** Feche todos os programas em execução no MARS antes de iniciar cada exercício abaixo!

### Questão 01)

#### Procedimento de simulação

- a) Abra o programa implementado no **exercício 1 do Laboratório Matrices** (Laboratório 05).
- b) Monte o programa (*Assemble*).
- c) No menu **Tools**, selecione **Data Cache Simulator**. Uma nova janela aparecerá no meio da tela.



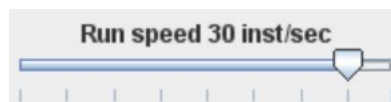
Esta é uma ferramenta do MARS que simulará o uso e o desempenho da memória cache quando um programa for executado no MIPS. Observe suas três seções principais:

- *Cache Organization*: Você pode usar as caixas de combinação para especificar como o cache será configurada para cada execução.

- *Cache Performance*: Com cada acesso à memória durante a execução do programa, o simulador determinará se esse acesso pode ou não ser satisfeito a partir da cache e atualizará a exibição do desempenho de acordo.
- *Runtime Log*: as mensagens de tentativa de acesso são mostradas no terminal.
- *Tool Control*: Esses botões executam funções de controle genéricas, conforme descrito por seus rótulos.

d) Clique no botão **Connect to MIPS**. Isso faz com que a ferramenta se registre como um “observador” da memória MIPS e, assim, responda durante a execução do programa.

e) De volta ao MARS, ajuste o controle deslizante **Run Speed** para 30 instruções por segundo. Ele está localizado no lado direito da barra de ferramentas. Isso retarda a execução para que você possa assistir à animação de desempenho da Cache:



f) No MARS, execute o programa usando o botão Executar da barra de ferramentas:



Observe a animação do desempenho da cache conforme ela é atualizada a cada acesso à memória do MIPS.

**Responda:** Qual foi a taxa final de acertos do cache?

**Obs.:** Perceba que, a cada falha, um bloco de 4 words é gravado na cache. Em passagem, os elementos da matriz são acessados na mesma ordem em que são armazenados na memória. Assim, cada falta de cache é seguida por 3 acertos, pois os próximos 3 elementos são encontrados no mesmo bloco de cache. Isso é seguido por outra falha quando o *Direct Mapping* é mapeado para o próximo bloco de cache e os padrões se repetem. Assim, 3 de cada 4 acessos à memória serão resolvidos na cache.

**Responda:** Dada essa explicação (acima), qual será a taxa de acertos se o tamanho do bloco for aumentado de 4 para 8 words? \_\_\_\_\_

**Responda:** E se for diminuído de 4 words para 2 words? \_\_\_\_\_

Verifique suas previsões modificando o tamanho do bloco e executando novamente o programa para 8 e depois 2 words por bloco.

NOTA: quando você modifica a organização do cache, os valores de desempenho são redefinidos automaticamente (você pode usar o botão **Reset** do painel **Tool Control**).

NOTA: Você também precisa reiniciar o programa no MIPS antes de poder executá-lo novamente. Para isso use o botão:



### Questão 2)

Repita a mesma simulação para o **exercício 2 do Laboratório Matriz em Cache**: com 08 blocos, tendo 4 words por bloco.

**Responda:** Qual foi o desempenho do cache para este programa? \_\_\_\_\_.

NOTA: O problema é que os locais de memória agora são acessados não sequencialmente como antes, mas cada acesso é 16 words além do anterior (circularmente). Com as configurações que usamos, não há dois acessos consecutivos à memória ocorrem no mesmo bloco, portanto, cada acesso é uma falha.

Siga os seguintes passos:

- Altere o tamanho do bloco para 16. Observe que isso redefinirá a ferramenta.
- Crie uma segunda instância do Cache Simulator selecionando novamente Data Cache Simulator no menu Tools. Ajuste os dois quadros para poder visualizar os dois ao mesmo tempo. Conecte a nova instância da ferramenta ao MIPS, altere seu tamanho de bloco para 16 e altere seu número de blocos para 16.
- Execute novamente o programa. **Responda:** Qual é o desempenho do cache da instância da ferramenta original? \_\_\_\_\_. O tamanho do bloco 16 não ajudou porque ainda havia apenas um acesso a cada bloco, o miss inicial, antes que o bloco fosse substituído por um novo. **Responda:** Qual é o desempenho do cache da segunda instância da ferramenta? \_\_\_\_\_. Nesse ponto, toda a matriz caberá no cache e, portanto, uma vez que um bloco é lido, ele nunca é substituído. Apenas o primeiro acesso a um bloco resulta em erro.

### Questão 3)

Considerando as duas implementações que você realizou para o **Laboratório Matriz em Cache**, qual das duas teve um melhor desempenho? Por quê? Justifique sua resposta.