Arthur de Lara Machado (22200348)
Felipe Fagundes Pacheco (22202615)

Relatório Laboratório 8

Exercício 1:

O programa solicita ao usuário o tamanho da matriz (MAX) e, em seguida, aloca dinamicamente espaço na memória para as matrizes A e B. Utilizando dois loops aninhados, o programa percorre cada elemento das matrizes, calcula os deslocamentos necessários, realiza a adição dos elementos correspondentes e armazena o resultado de volta na matriz A.

Exercício 2:

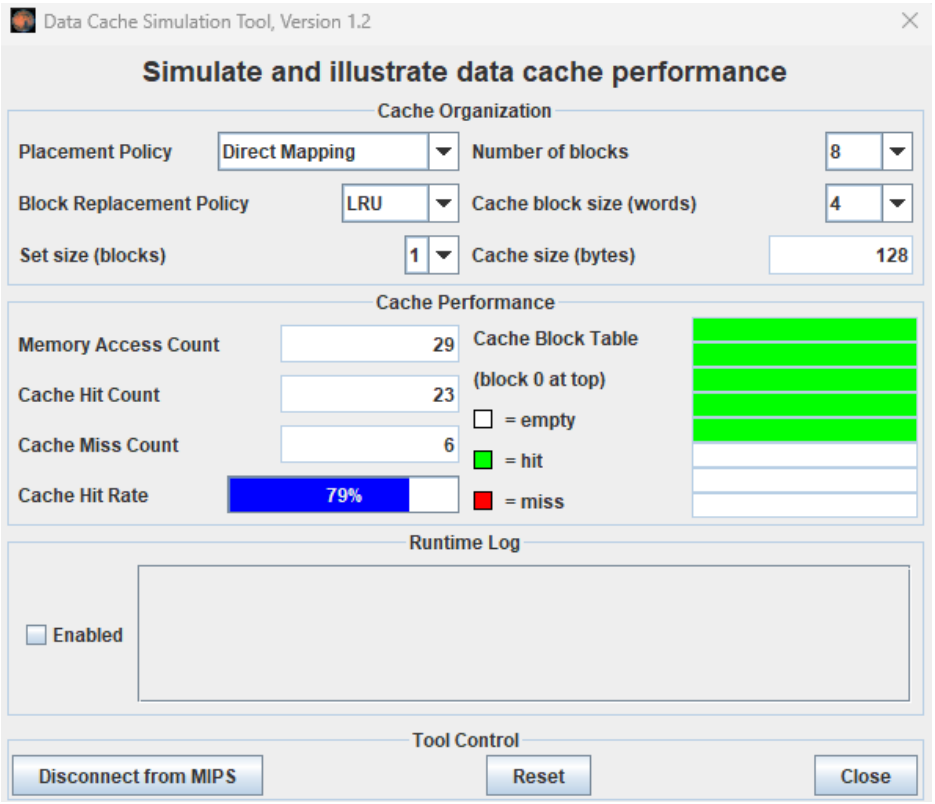
Este programa eficientemente realiza a adição de matrizes usando a técnica de cache blocking. Após obter do usuário os tamanhos da matriz quadrada (MAX) e do bloco (block_size), ele aloca dinamicamente memória para as matrizes A e B. Utilizando loops aninhados, o programa percorre os blocos da matriz, otimizando o acesso à memória. Cálculos de deslocamento são aplicados para acessar os elementos corretos nas matrizes, onde a adição é executada, atualizando a matriz A.

Exercício 3:

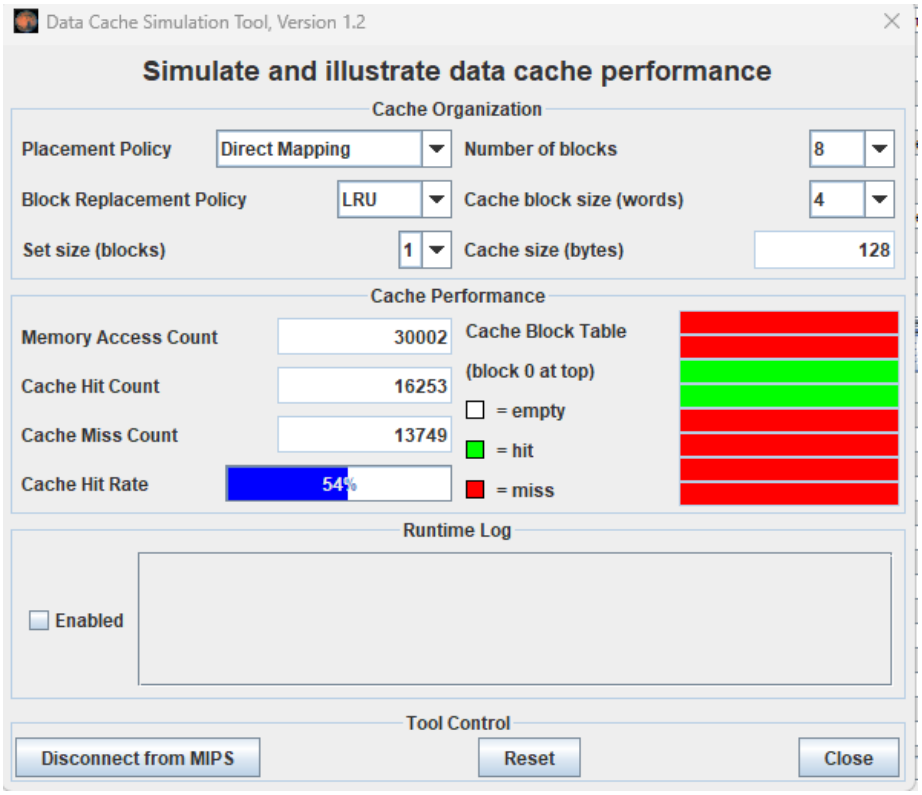
Foram realizados dois testes em cada programa, para serem feitas as simulações, foram realizados testes em uma matriz pequena (3x3) (1 block_size no segundo programa) e outra média (100x100) (10 block_size no segundo programa), uma matriz grande acabou não sendo possível, até mesmo crashando o MARS para tamanhos maiores, como 1000x1000, então os testes foram esses.

Primeiro programa:

Tamanho pequeno:

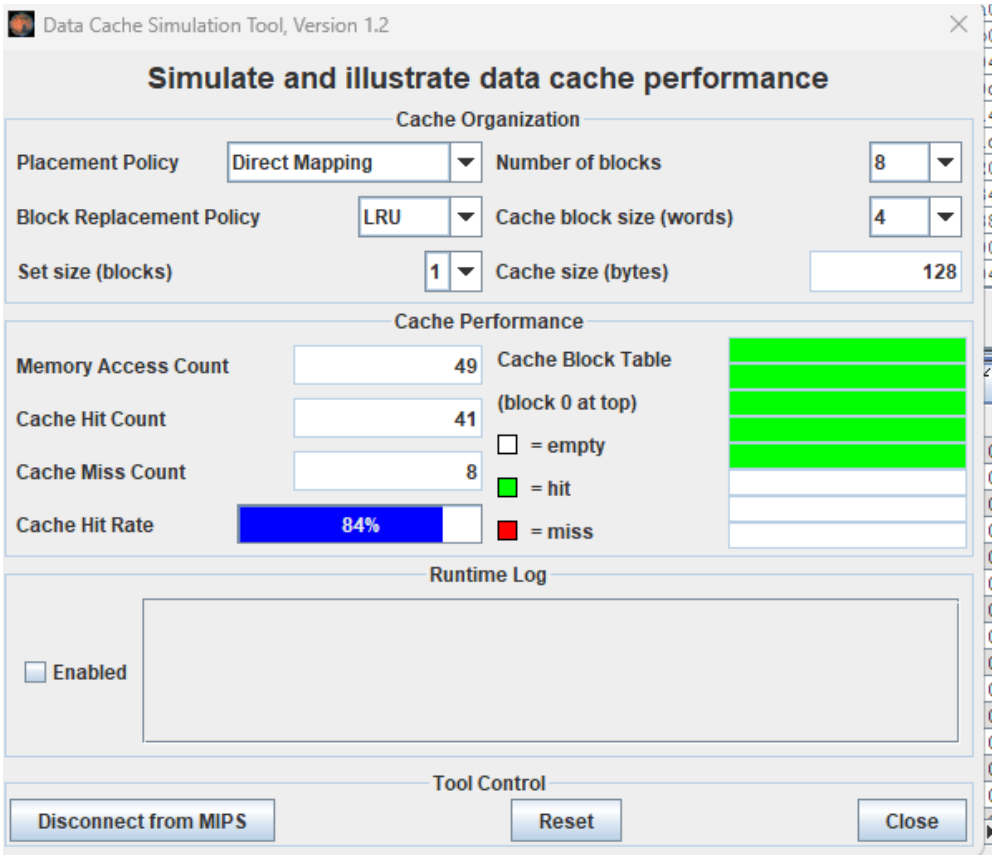


Tamanho médio:

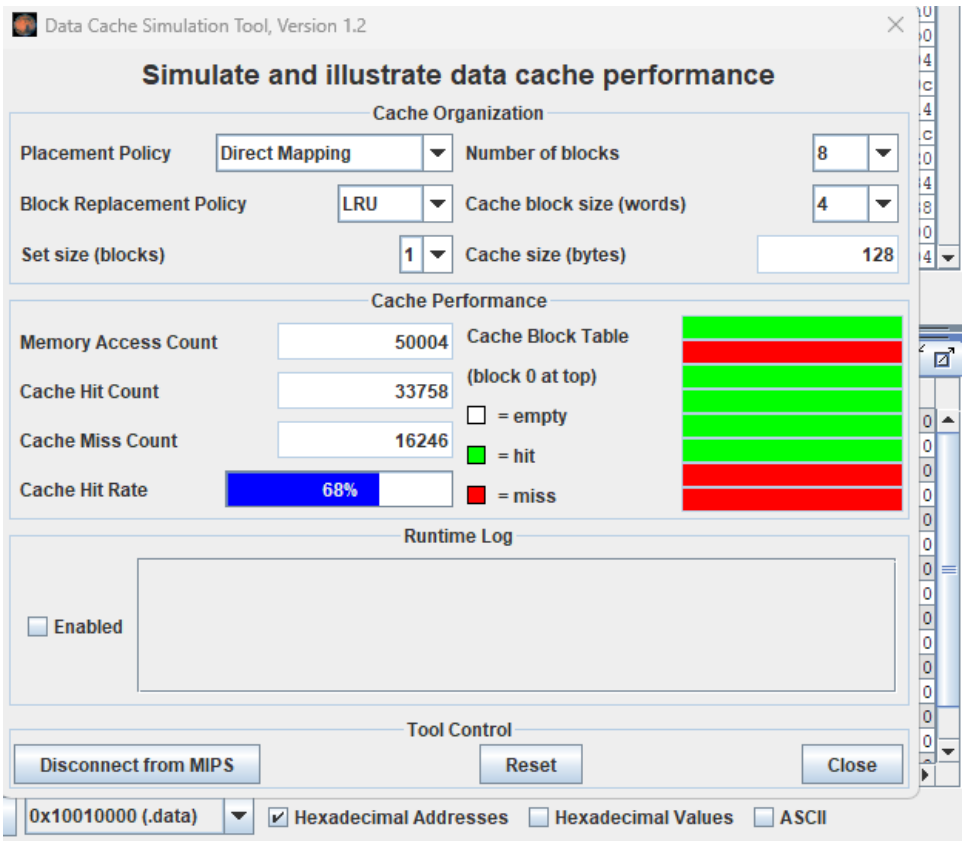


Segundo Programa:

Tamanho pequeno:



Tamanho médio:



- No tamanho pequeno, o programa com cache blocking obteve uma taxa de acertos na cache ainda melhor (84% contra 79%). Isso sugere que a técnica de cache blocking está proporcionando benefícios mesmo para matrizes pequenas.
- No tamanho médio, a taxa de acertos com cache blocking é menor (68% contra 54%). Isso pode indicar que a eficácia do cache blocking pode depender do tamanho específico da matriz e do padrão de acesso à memória.
- Ambos os programas mostram uma diminuição na taxa de acertos à medida que o tamanho da matriz aumenta, o que é esperado devido a um maior número de acessos à memória e à complexidade do algoritmo.
- Outra consideração que podemos tirar, é na velocidade de execução da simulação, que por haver um tamanho de bloco, o segundo programa demorou um pouco mais que o primeiro, porém é um tempo que é válido, pois o seu desempenho é otimizado