

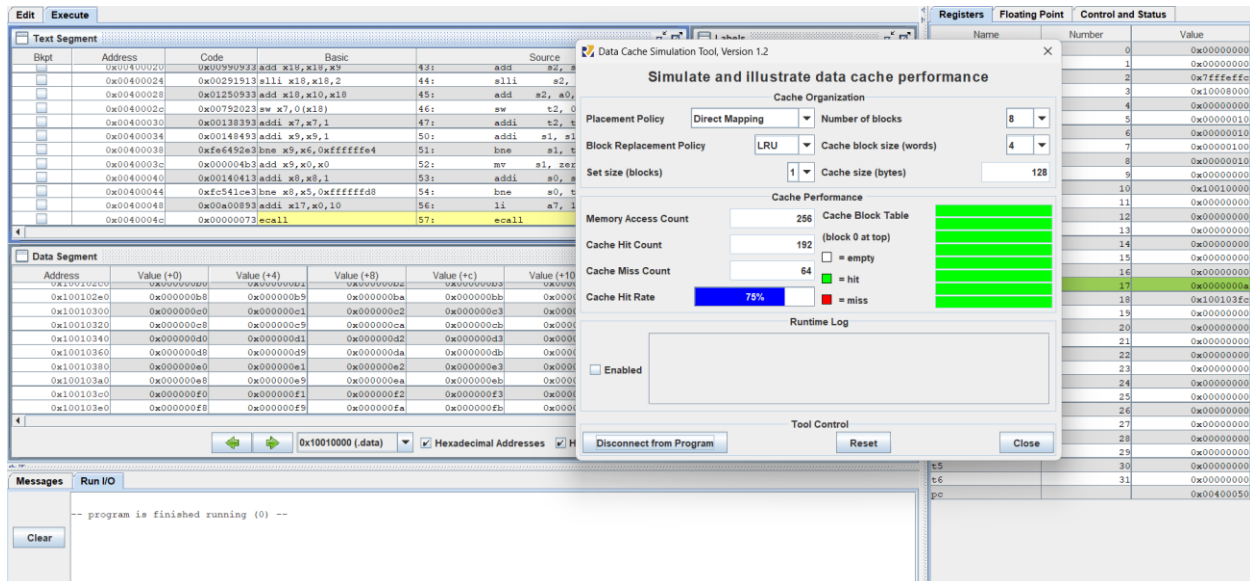
Bài 12. Bộ nhớ đệm nhanh – Cache memory

Họ và tên: Nguyễn Thành Duy

MSSV: 20235696

Assignment 1 – Sử dụng công cụ Data Cache Simulator

7. Tỷ lệ cache hit sau khi chạy chương trình? - 75%



8.

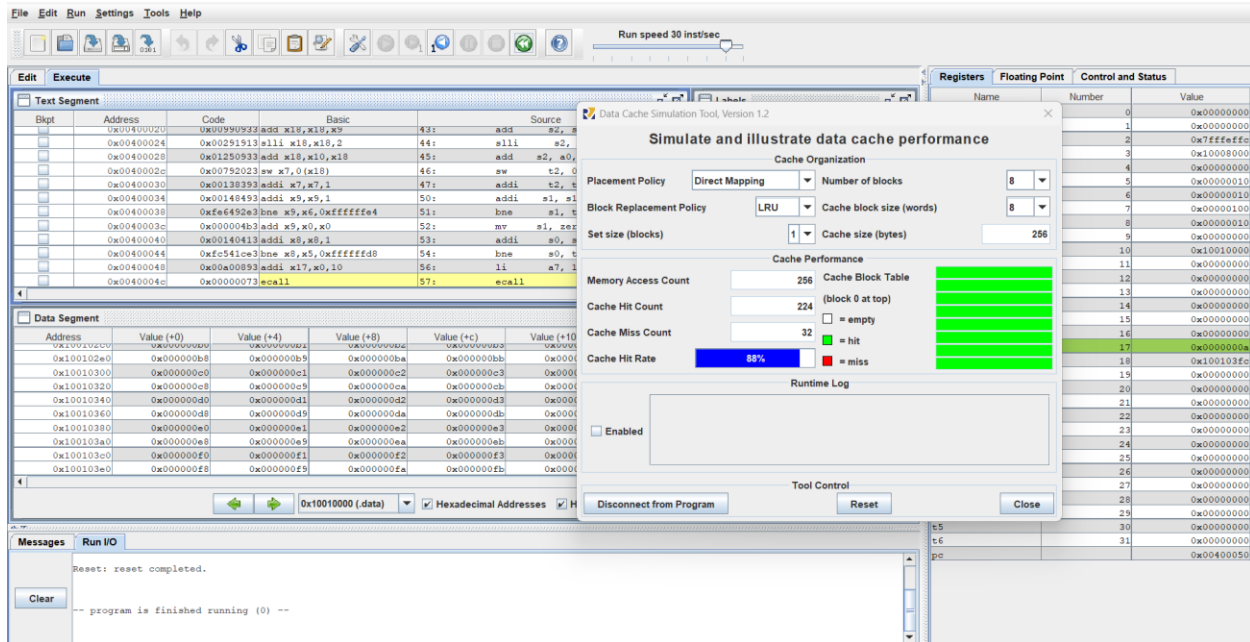
- Tỷ lệ cache hit nếu kích thước khối block tăng từ 4 thành 8 từ nhớ: **88%**

Giải thích:

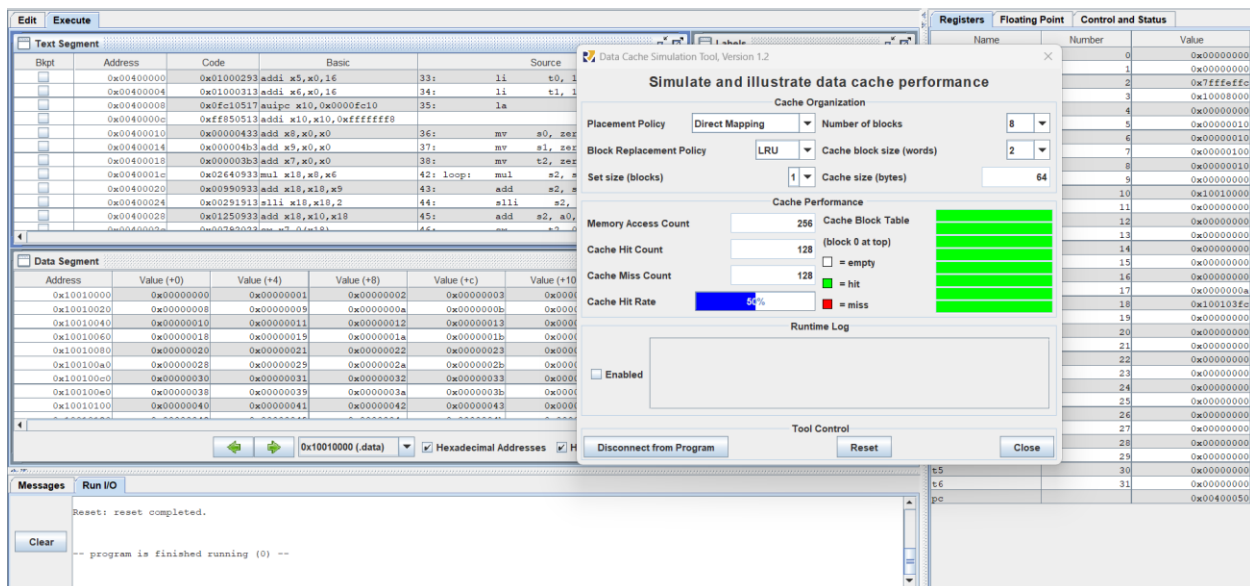
- Với 4 từ nhớ, sau mỗi lần cache miss, ta có 3 lần cache hit (tỷ lệ **75%**)
 - ➔ 8 từ nhớ, ta có 7 lần cache hit (tăng tỷ lệ lên **88%**)
 - ➔ 2 từ nhớ, ta có 1 lần cache hit (tỷ lệ **50%**)

9.

- 8 từ nhớ:

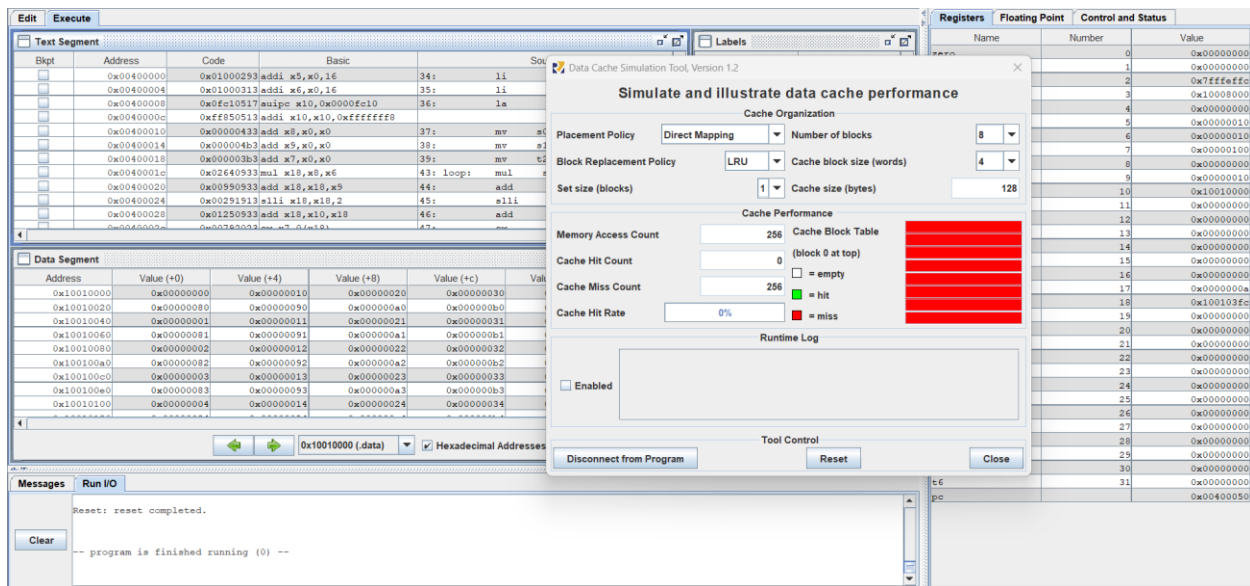


- 2 từ nhớ:



10.

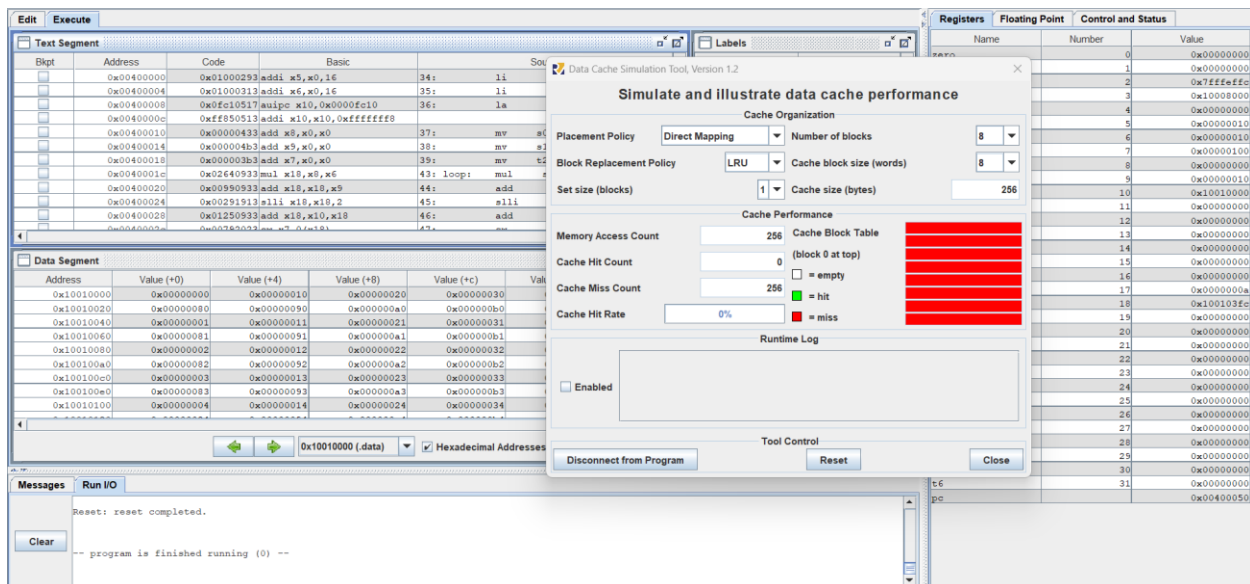
- Tỷ lệ cache hit sau khi chạy chương trình? 0%

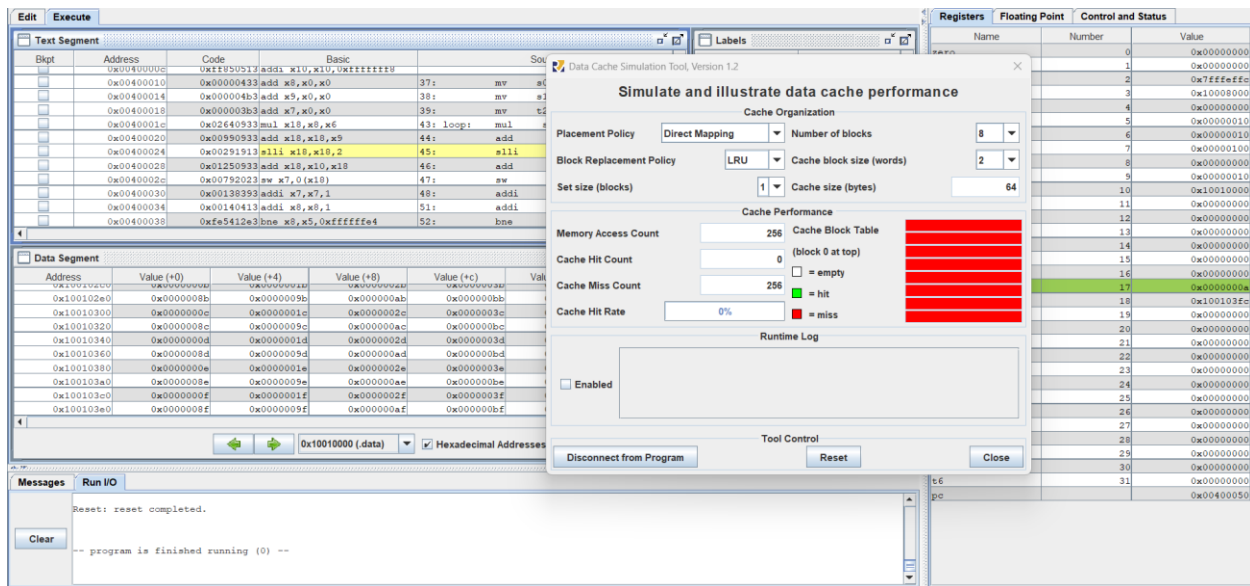


- Tỷ lệ cache hit nếu kích thước khối block tăng từ 4 thành 8 từ nhớ? **0%**
- Tỷ lệ cache hit nếu kích thước khối block giảm từ 4 thành 2 từ nhớ? **0%**

Giải thích:

- Với cách duyệt theo cột, các phần tử của ma trận không được truy cập theo cùng thứ tự lưu trữ trong bộ nhớ nên khi cache miss ta không có lần cache hit nào nên tỷ lệ cache hit rate là 0%





11. Hiệu năng truy nhập bộ nhớ cache đối với chương trình này là bao nhiêu? - 0%

14. Hiệu năng truy nhập được tính trên cửa sổ cũ là bao nhiêu? - 0%

- Hiệu năng truy nhập được tính trên cửa sổ mới là bao nhiêu? – 94%

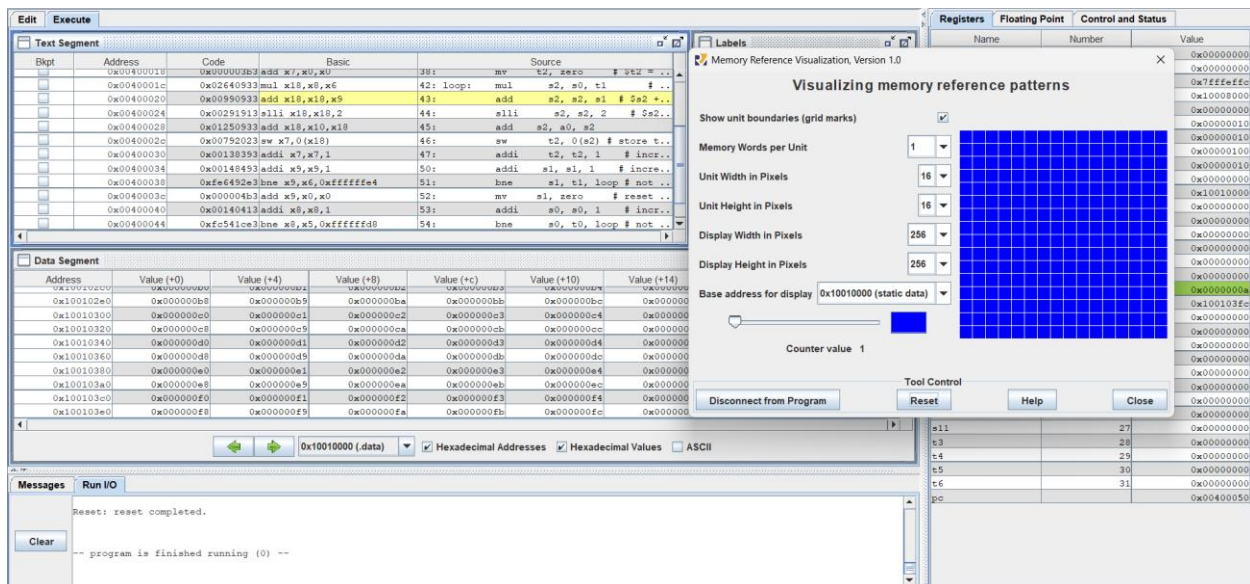
Giải thích: Trong trường hợp này, toàn bộ ma trận sẽ vừa với bộ nhớ cache và một khi block được đọc, nó sẽ không bị thay thế bởi block khác. Chỉ có lần truy nhập block đầu tiên trả về kết quả là cache miss.

Assignment 2 – Sử dụng công cụ Memory Reference Visualization

row-major.asm

Kết quả:

- Ta thấy tất cả các ô lưới hay một từ nhớ đều có **màu xanh** (tức được truy cập 1 lần).



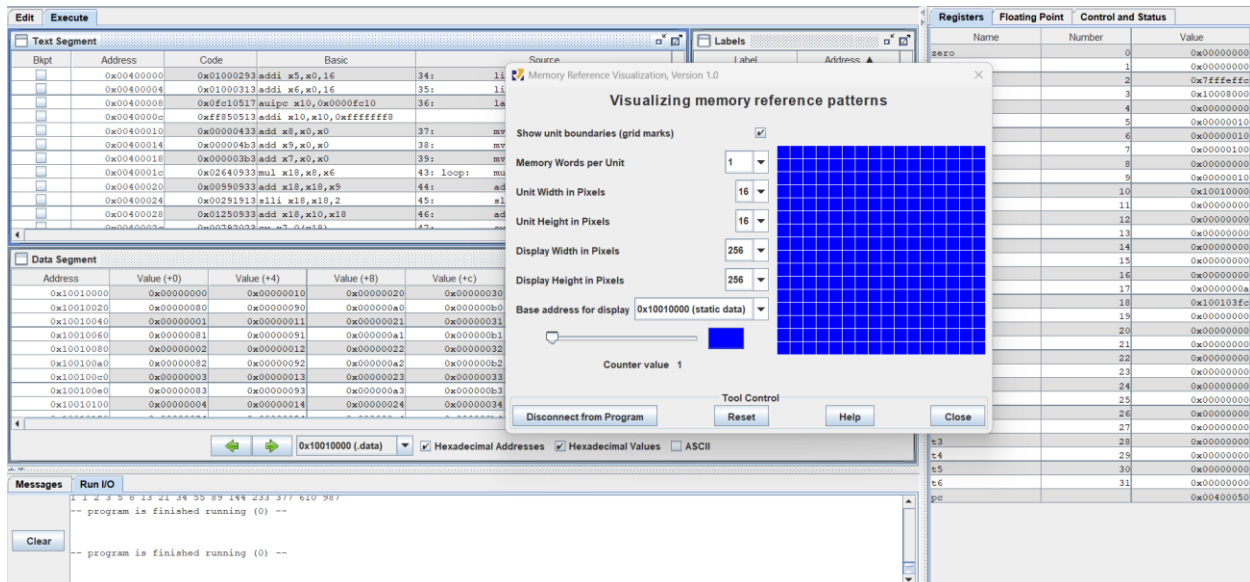
Giải thích:

- ⇒ Với cách duyệt theo dòng và với mỗi dòng duyệt qua các phần tử vậy nên các từ nhớ được truy cập liên tiếp 1 lần.

column-major.asm.

Kết quả:

- Tất cả các ô lưới hay một từ nhớ đều có **màu xanh** (tức được truy cập 1 lần).

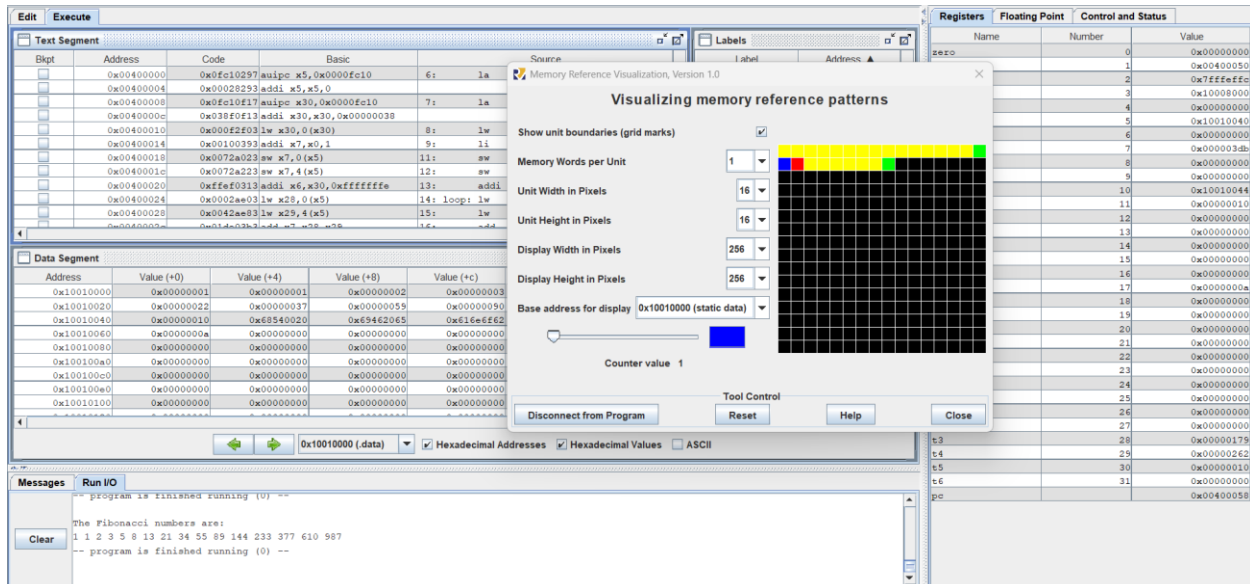


Giải thích:

⇒ Với cách duyệt theo từng cột thì mỗi phần tử hay từ nhớ cũng đều được truy cập 1 lần nên tất cả các ô lưới đều có màu xanh.

fibonacci.asm

Kết quả:



Giải thích:

- Ở dòng đầu tiên của Tool: Đây là các phần tử của mảng gồm 16 phần tử
 - ⇒ Từ phần tử 0-15 có màu vàng tức được truy cập 3 lần, phần tử cuối màu xanh tức được truy cập 1 lần:
 - + Phần tử thứ nhất
 - Một lần ở câu lệnh gán giá trị bằng 1
 - Một lần gọi lại để tính giá trị ở phần tử kế tiếp
 - Một lần để lấy địa chỉ của mảng
 - + Phần tử 2 - 14 :
 - Một lần ở câu lệnh gọi lấy giá trị thông qua câu lệnh lw
 - Một lần gọi để tính giá trị
 - Một lần gọi lại để tính giá trị ở phần tử kế tiếp
 - + Phần tử cuối cùng:
 - Một lần gọi lại để tính giá trị- Ở dòng thứ hai của Tool: Đây là các biến tạm thời trong bộ nhớ đệm dùng trong print_fibs