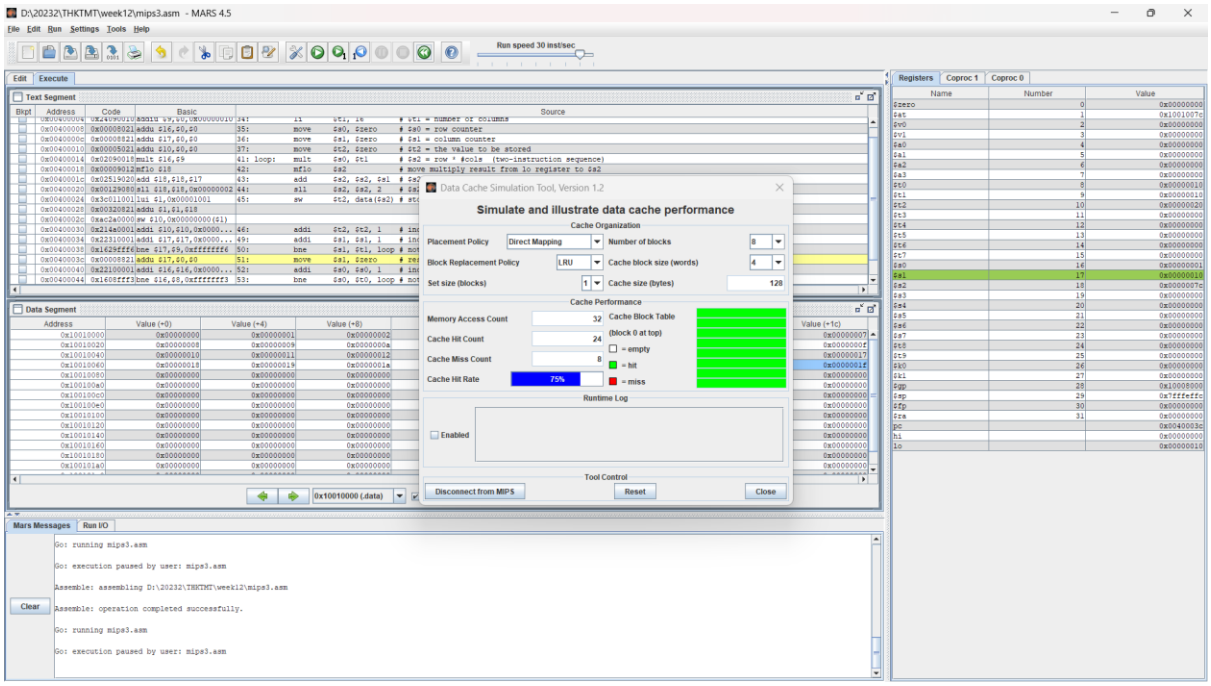
THỰC HÀNH KIẾN TRÚC MÁY TÍNH

WEEK 12

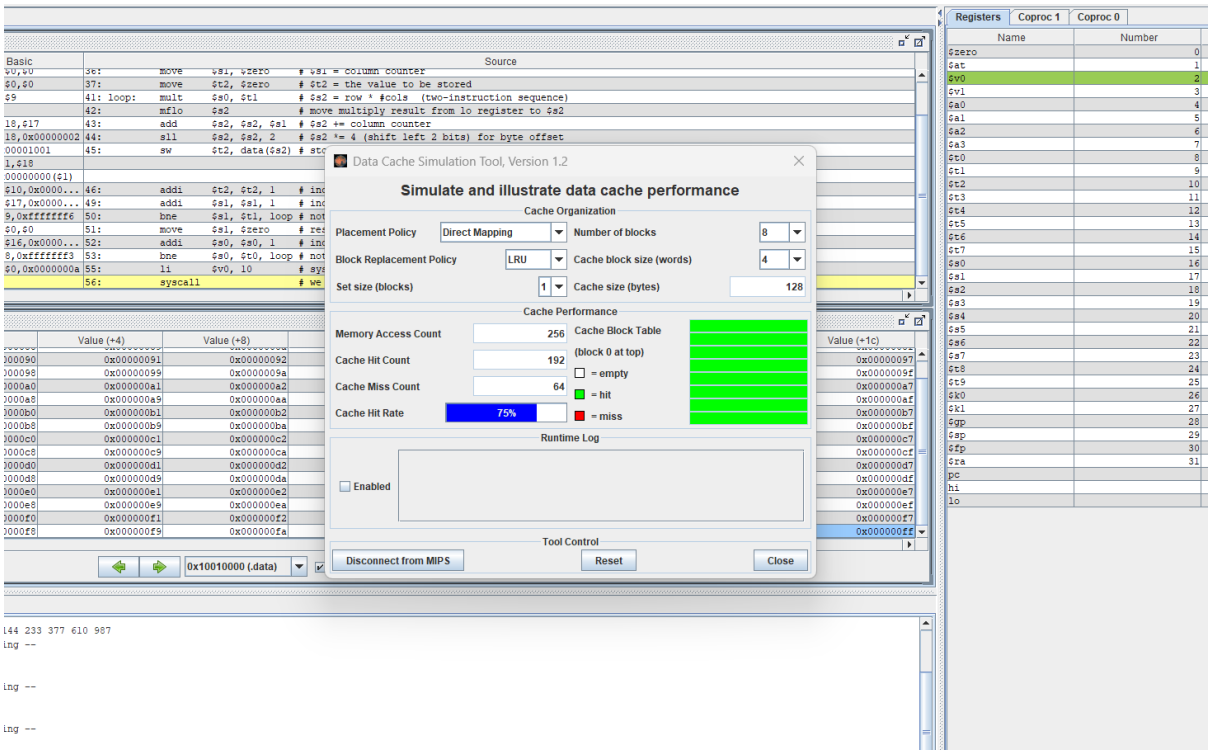
BÙI QUANG HƯNG – 20225849

Assignment 1:

- Thực hiện từ bước 1 đến bước 7 với chương trình row-major:



- Kết quả cuối cùng

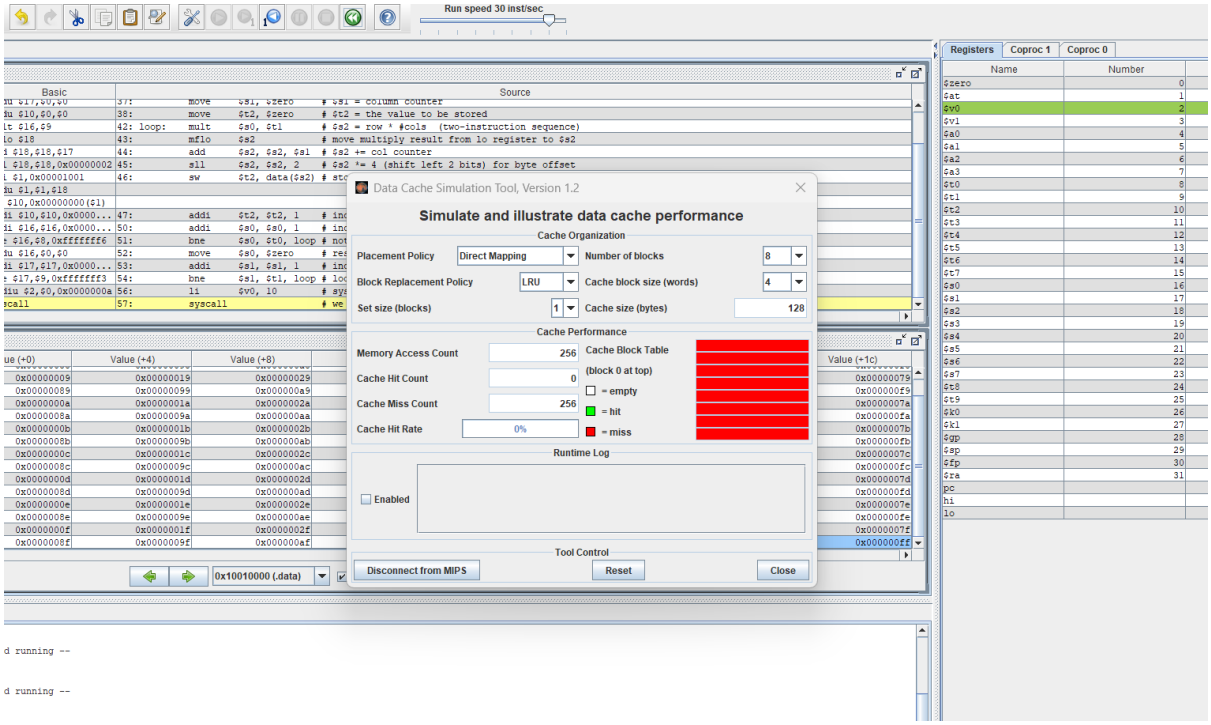


- Tỉ lệ cache hit rate cuối cùng là 75%. Vì với mỗi lần bỏ lỡ, một khối 4- word được ghi vào bộ đệm. Trong một đường truyền chính theo hàng, các phần tử của ma trận là được truy cập theo cùng thứ tự chúng được lưu trữ trong bộ nhớ. Do đó, mỗi lần bỏ lỡ bộ đệm là theo sau là 3 lần truy cập vì 3 phần tử tiếp theo được tìm thấy trong cùng một khối bộ đệm. Tiếp theo là một lỗi khác khi ma trận trực tiếp ánh xạ tới khối bộ đệm tiếp theo và sau đó lặp đi lặp lại chính nó. Vì vậy, 3 trong số 4 lần truy cập bộ nhớ sẽ được giải quyết trong bộ đệm.

- Dự đoán, khi tăng blocksize từ 4 lên 8 thì tỉ lệ Cache hit rate là 87,5%, còn khi giảm blocksize từ 4 xuống 2 thì thỉ lệ Cache hit rate còn 50% .

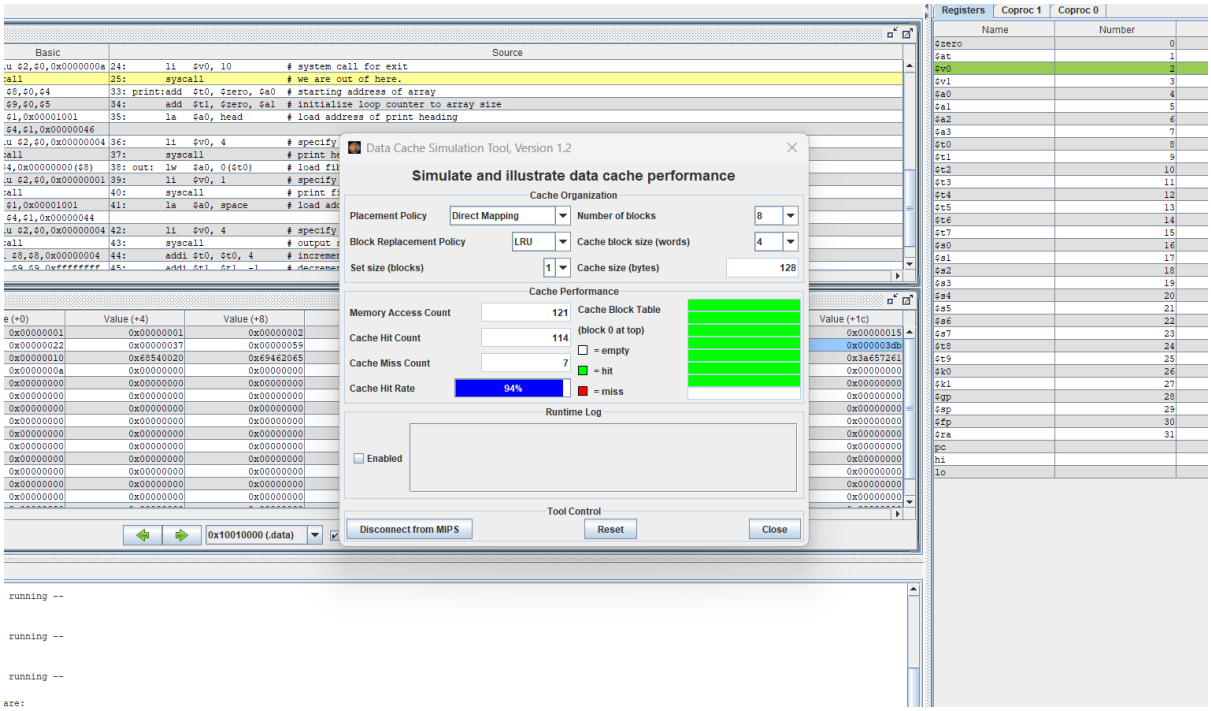
- Hiệu suất bộ đệm cũng bằng tỉ lệ Cache hit rate và bằng 75%.

- Trường hợp chạy chương trình column:



- Khi chạy chương trình column, tỉ lệ Cache hit rate là 0%, suy ra hiệu suất bộ nhớ đệm cũng là 0%

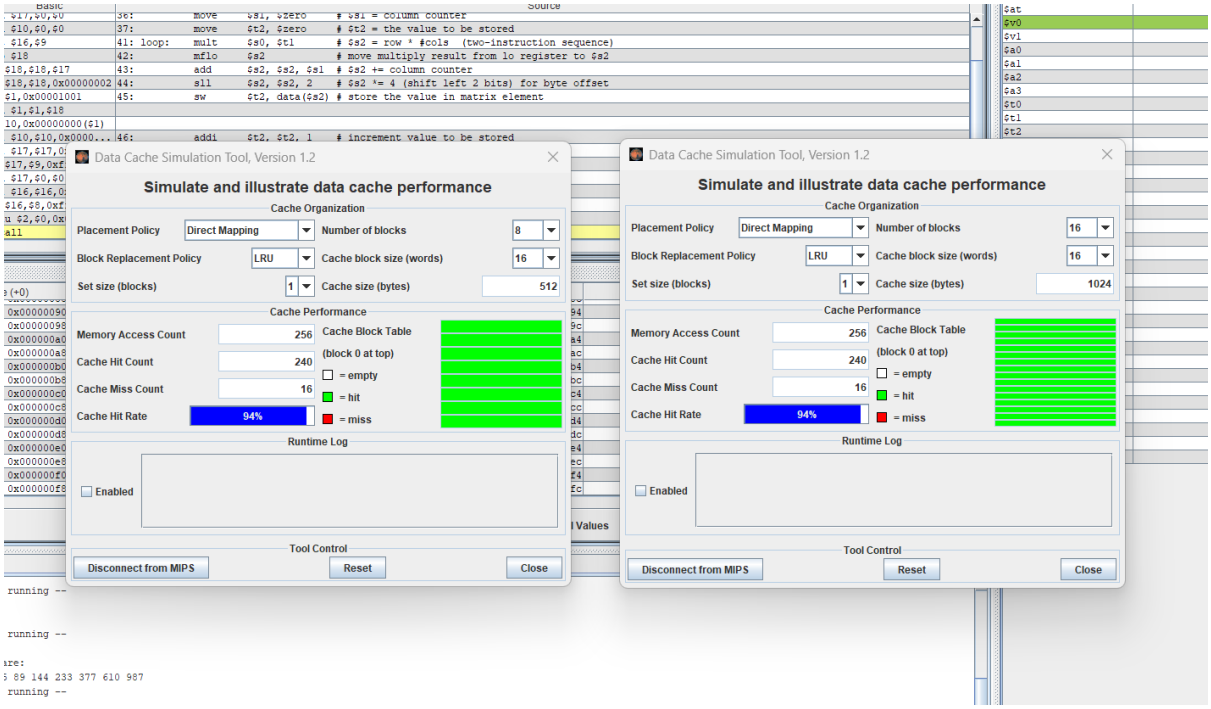
- Trường hợp chạy chương trình Fibonacci:



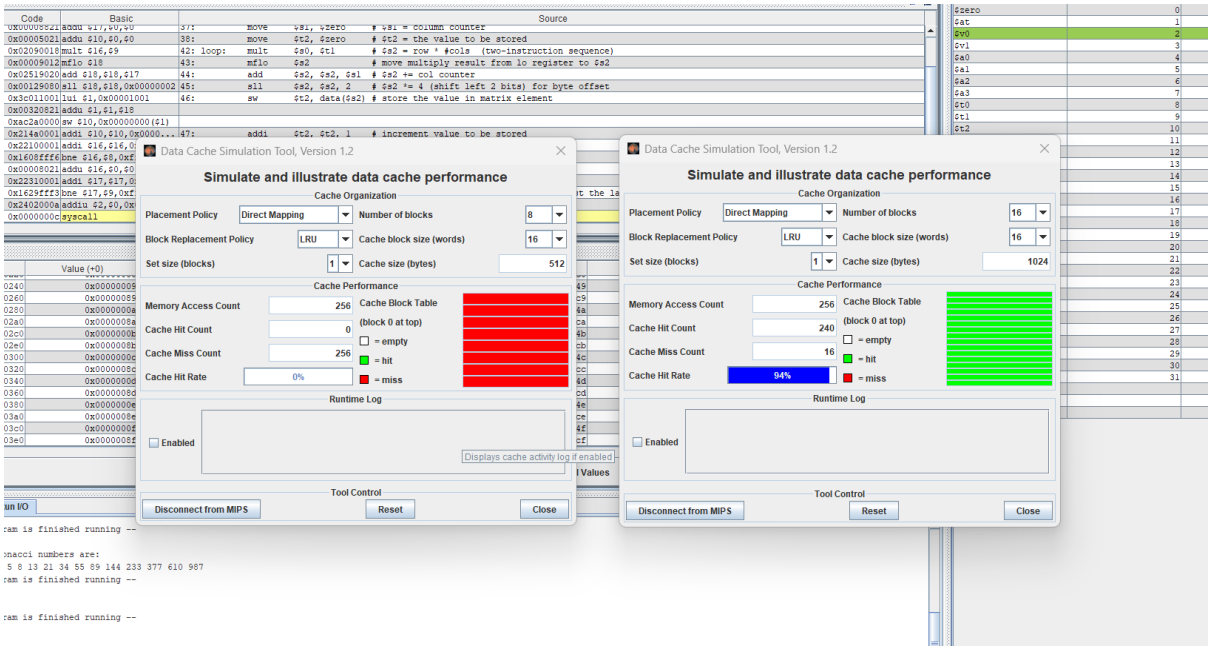
- Khi chạy chương trình Fibonacci, hiệu suất bộ nhớ đệm là 94%.

- Thực hiện từ bước 13 đến bước 15 (Trong các ảnh kết quả, cửa sổ dât cache bên trái là bản gốc, cửa số data cache bên phải là bản mới):

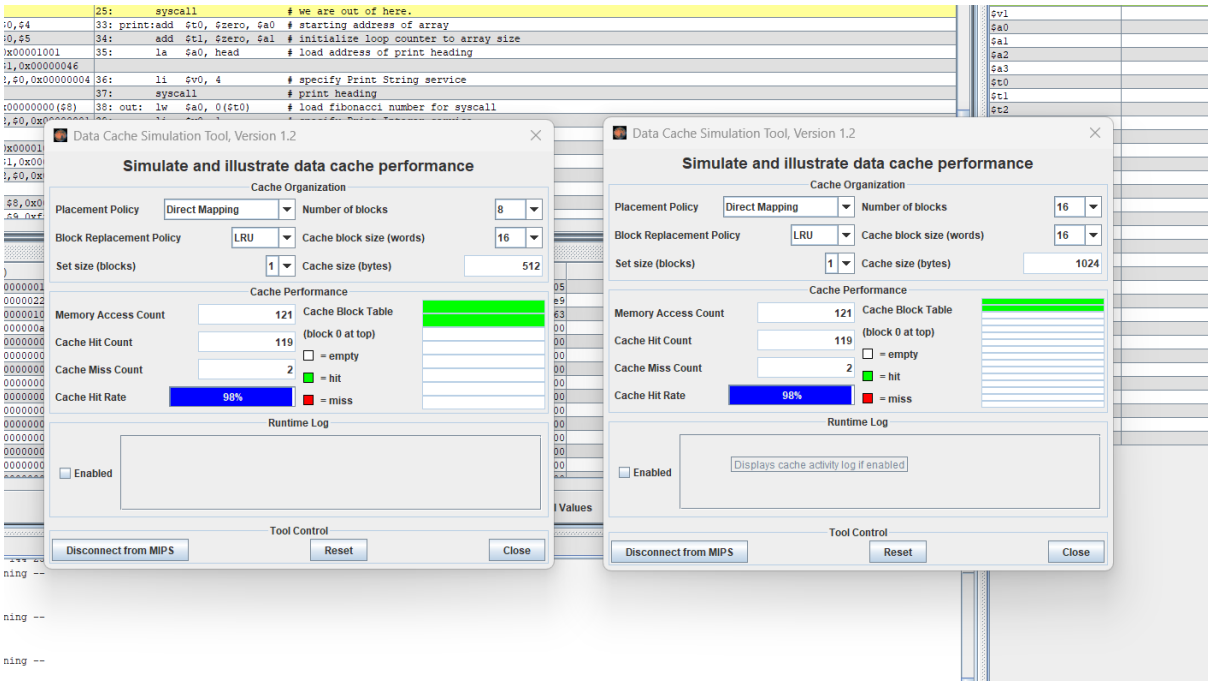
+ Khi chạy chương trình row-major:



+ Khi chạy chương trình column-major:



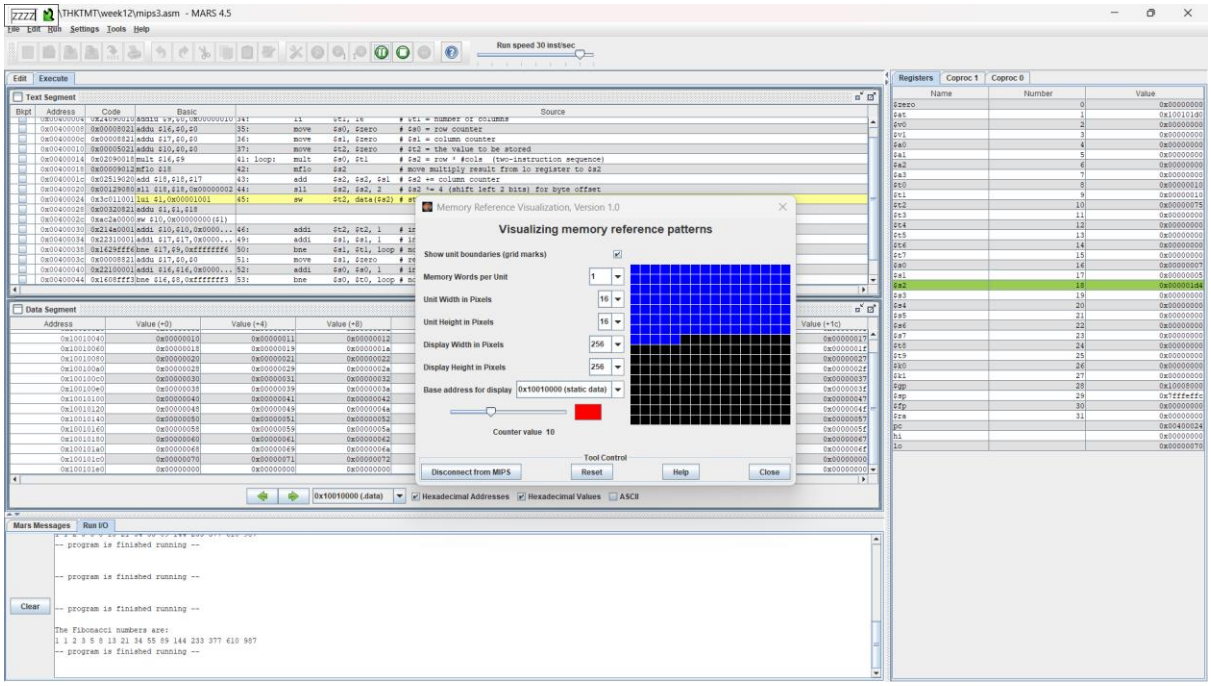
- Khi chạy chương trình Fibonacci:



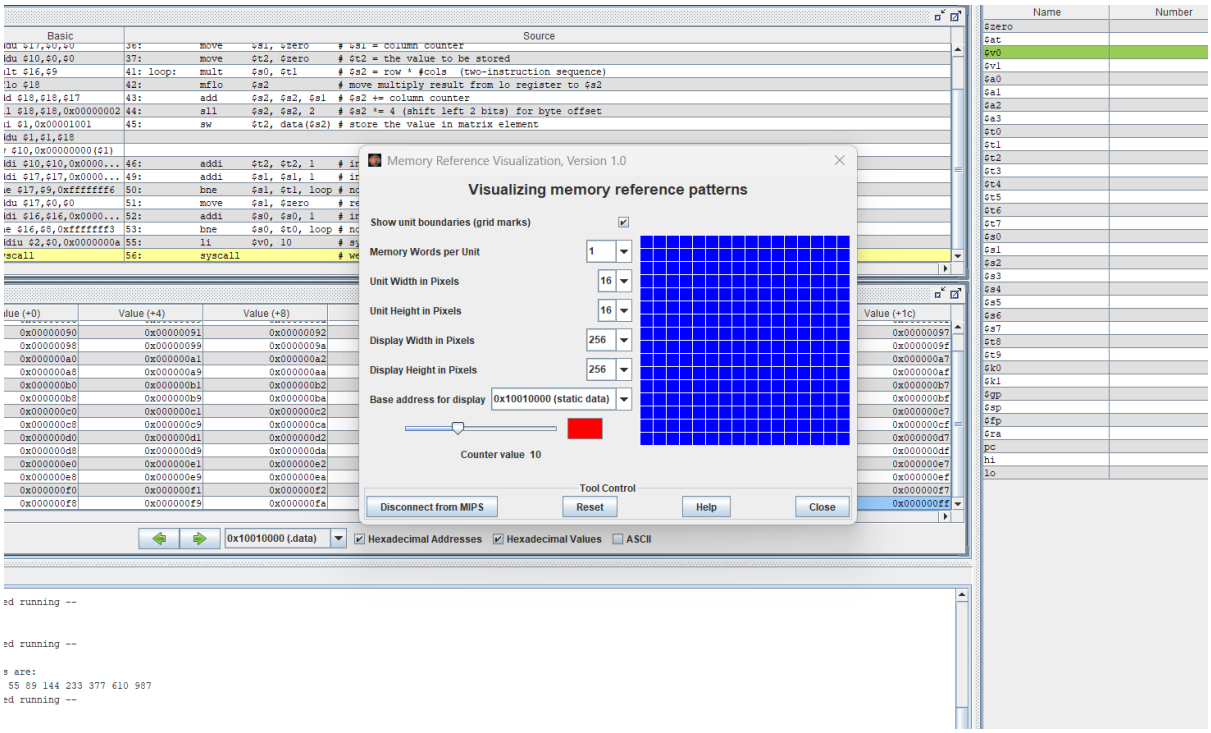
Assignment 2:

- Với chương trình row-major:

+ Khi đang chạy chương trình:

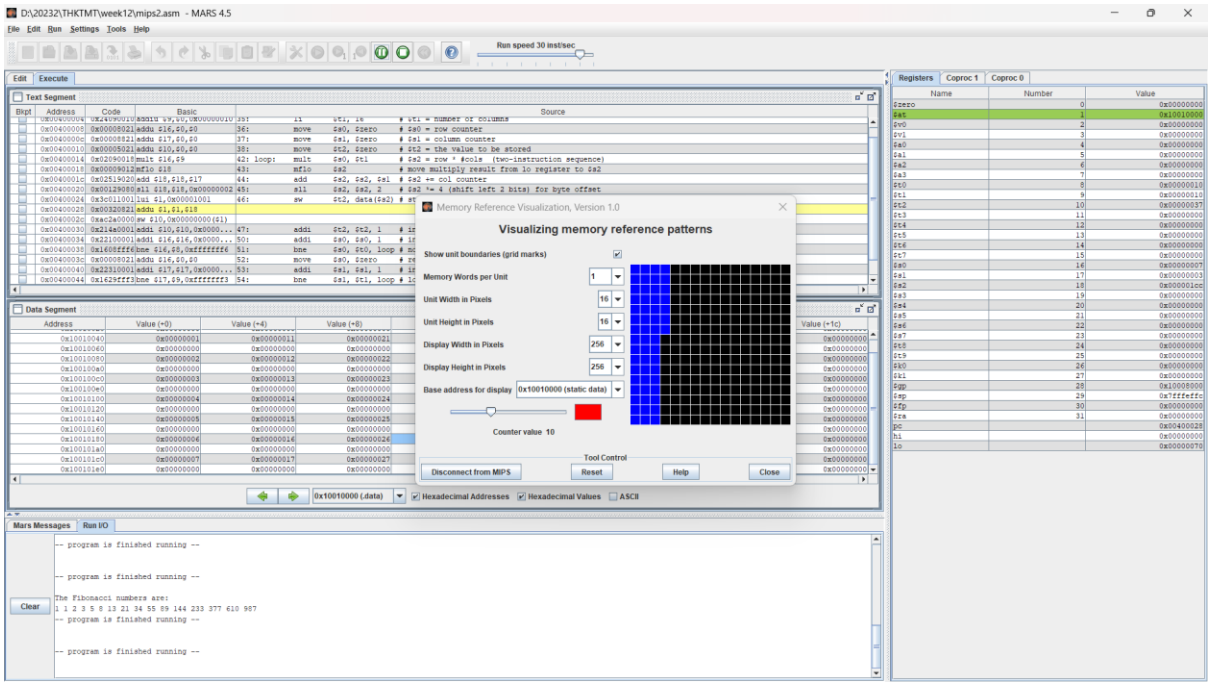


+ Khi đã chạy xong chương trình:

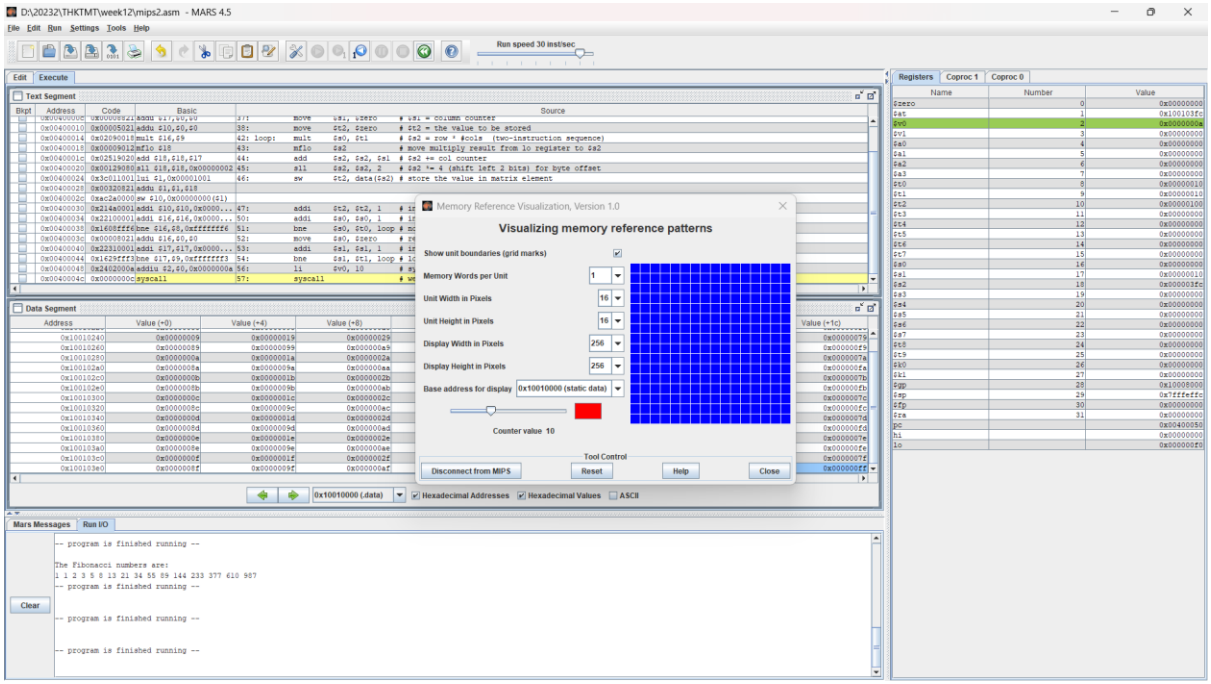


- Với chương trình column-major:

+ Khi chương trình đang chạy:

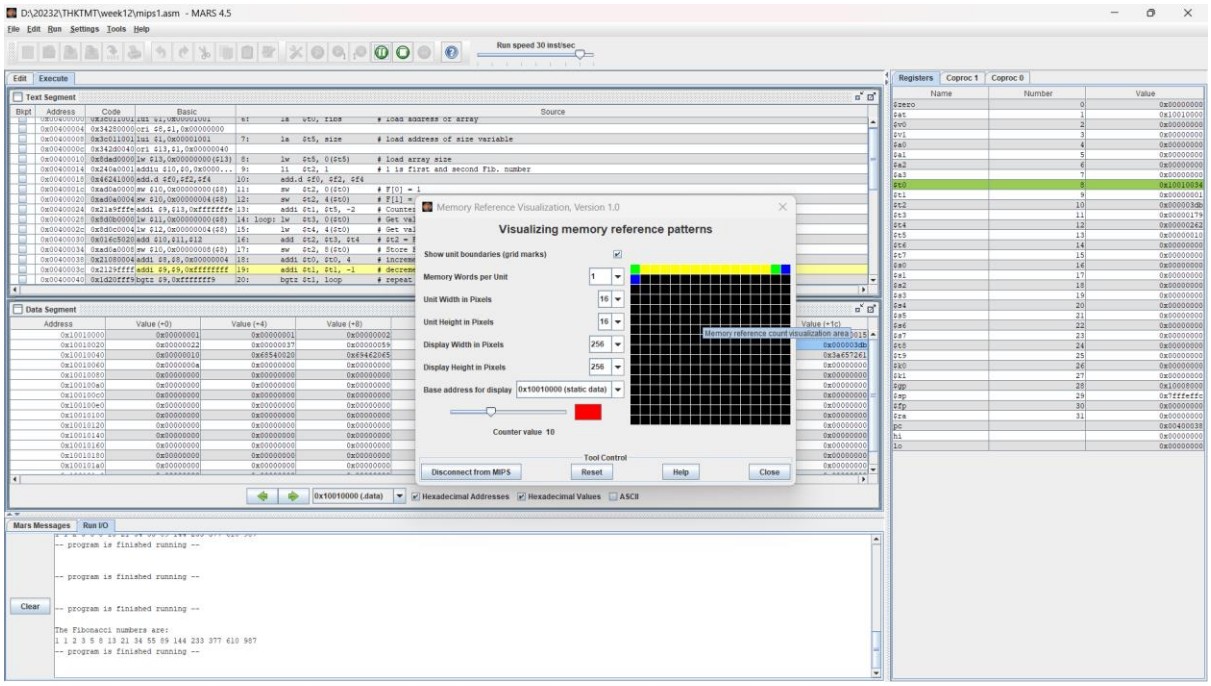


+ Khi đã chạy xong chương trình



- Với chương trình Fabonacci:

+ Khi chương trình đang chạy:



+ Khi chương trình đã chạy xong: