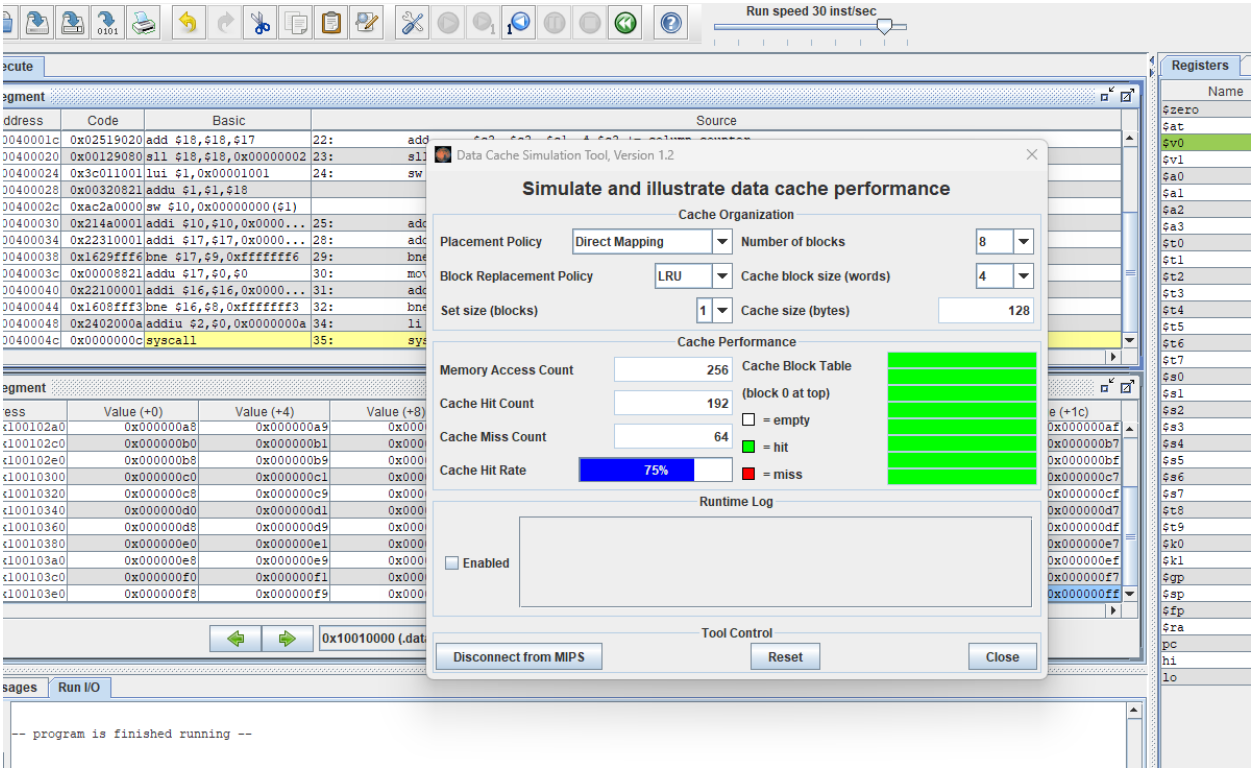
Thực hành Kiến trúc máy tính tuần 12

Họ tên: Đỗ Hoàng Minh Hiếu

MSSV: 20225837

BÀI 1:

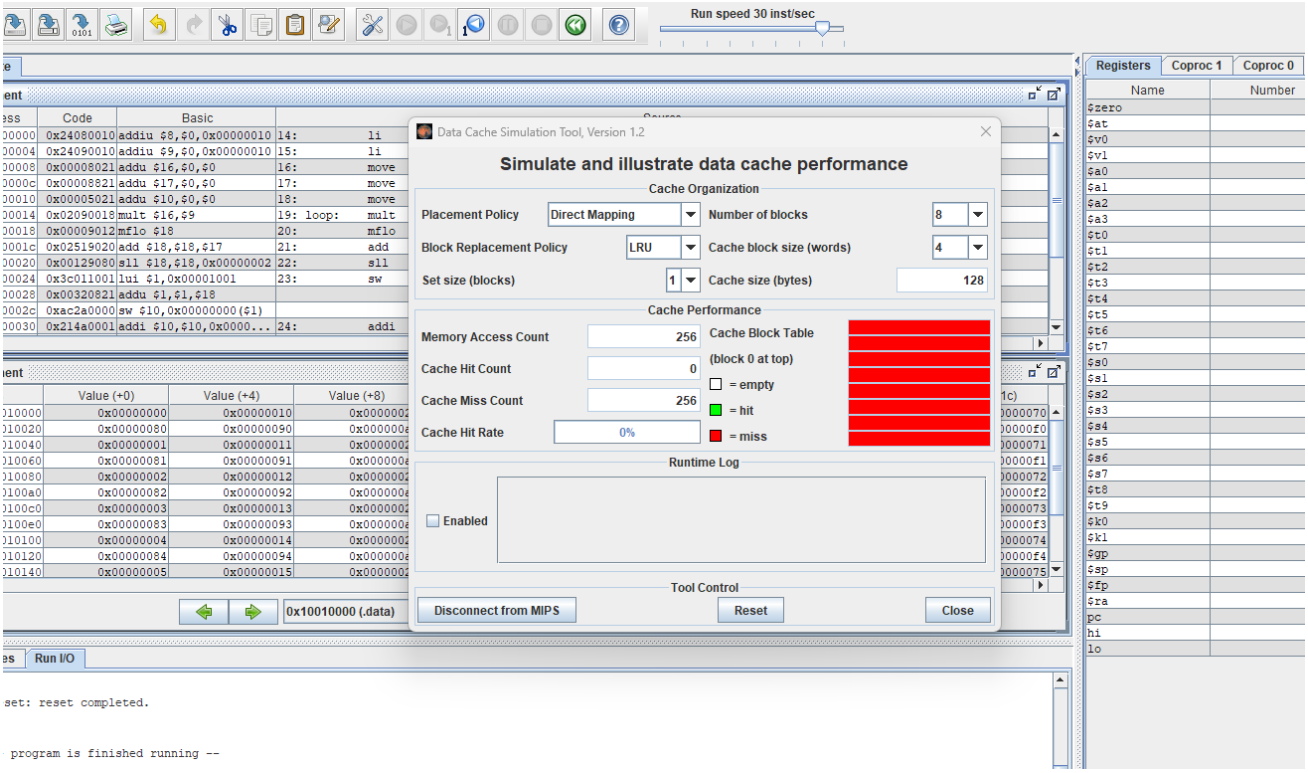
Kết quả thực thi chương trình row:



8. Tỉ lệ cache hit rate cuối cùng: 75%. Vì với mỗi lần miss, một khối 4 word sẽ được ghi vào bộ đệm. Trong mỗi lần duyệt 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, sau mỗi lần cache miss là 3 lần hit vì 3 phần tử tiếp theo được tìm thấy trong cùng một khối bộ đệm. Tiếp theo là lần miss khác khi ánh xạ trực tiếp 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.

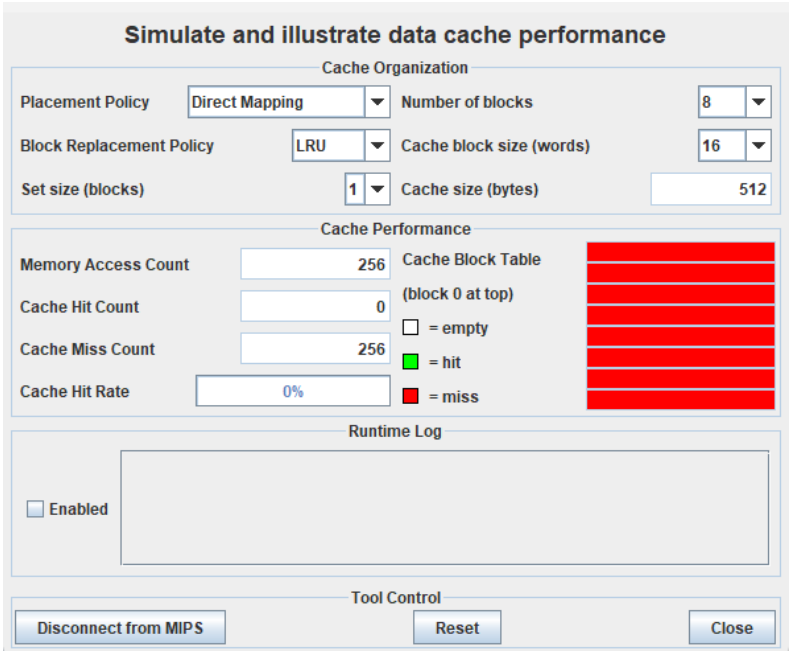
9. Dự đoán, khi tăng block size từ 4 lên 8 thì tỉ lệ Cache hit rate là 87,5%, còn khi giảm block size từ 4 xuống 2 thì thỉ lệ Cache hit rate còn 50%

- Trường hợp chạy chương trình column:



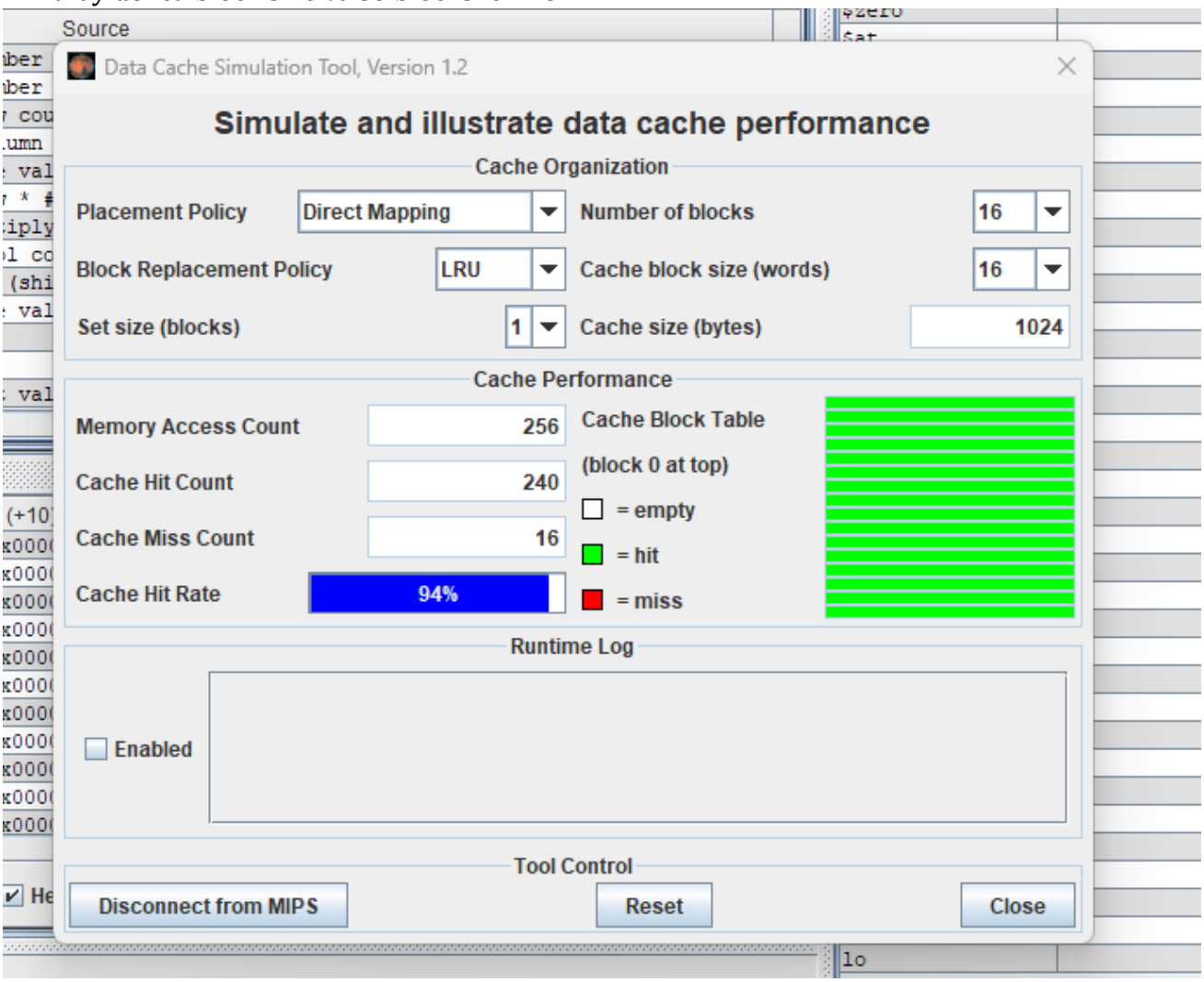
12. Hiệu suất bộ nhớ đệm là: 0%. Vấn đề ở đây là các vị trí bộ nhớ hiện được truy cập không tuần tự như trước, nhưng mỗi lần truy cập dài hơn lần truy cập trước đó 16 từ (theo vòng tròn). Với cài đặt mà chúng tôi đã sử dụng, không có hai lần truy cập bộ nhớ liên tiếp nào xảy ra trong cùng một khối nên mọi lần truy cập đều bị miss.

Nếu chỉ thay đổi block size lên 16:



Hiệu suất vẫn là 0%. Block size 16 không giúp ích gì vì vẫn chỉ có một quyền truy cập vào mỗi khối, lỗi ban đầu, trước khi khối đó được thay thế bằng khối mới.

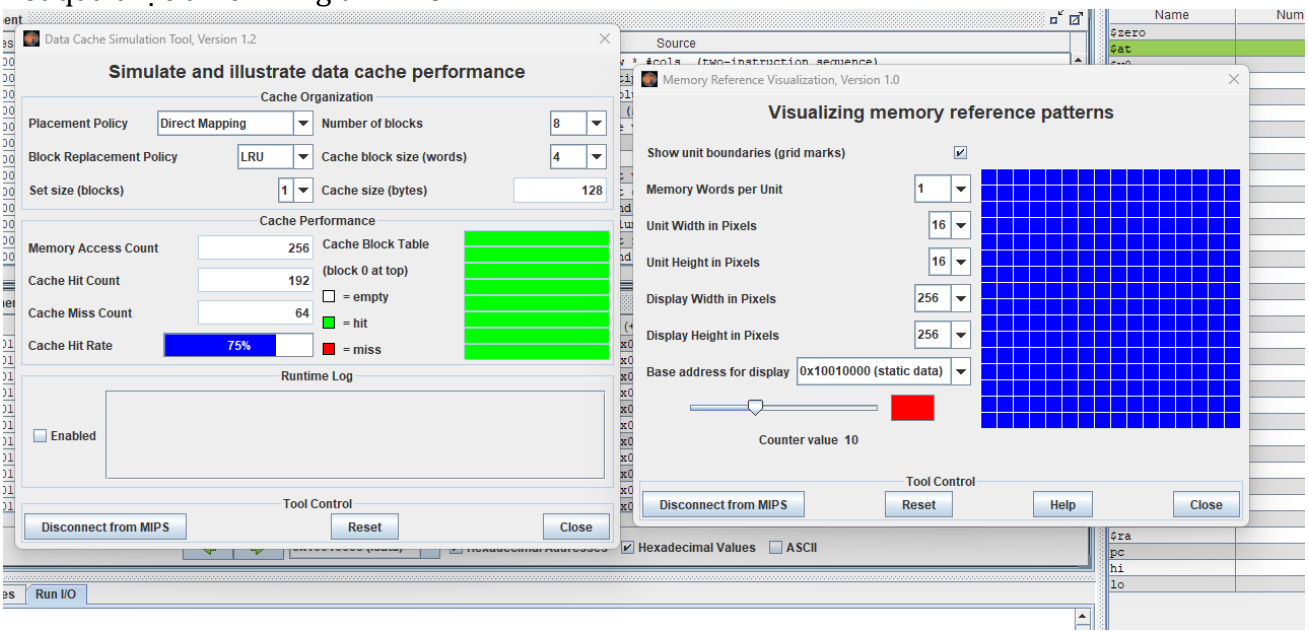
Khi thay đổi cả block size và số blocks lên 16:



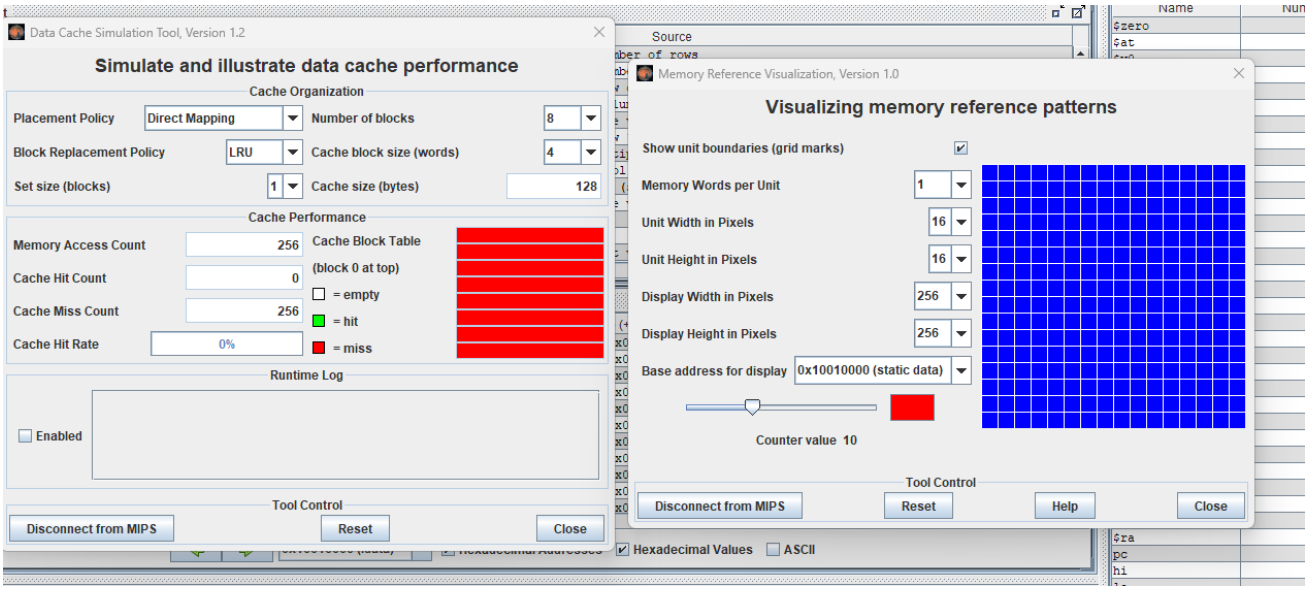
Hiệu suất trong trường hợp này lên đến 94%. Vì tại đây, toàn bộ ma trận sẽ được đưa vào bộ đệm và do đó, khi một khối được đọc vào thì khối đó sẽ không bao giờ được thay thế. Chỉ có lần truy cập đầu tiên vào một khối mới bị miss.

BÀI 2:

Kết quả thực thi chương trình row:



Kết quả thực thi chương trình column:



Kết quả thực thi chương trình Fibonacci: