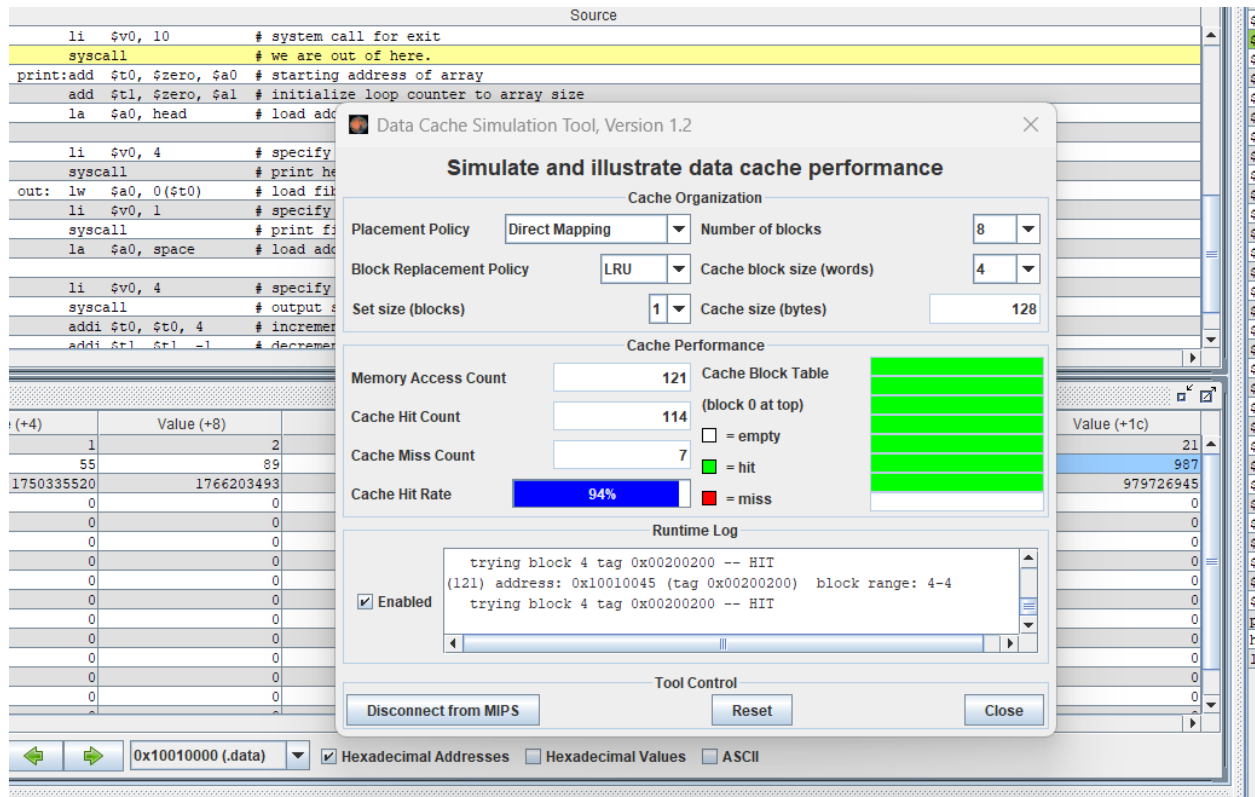


## Bài thực hành 12: Cache Memory

Nguyễn Văn Hiếu-20225717

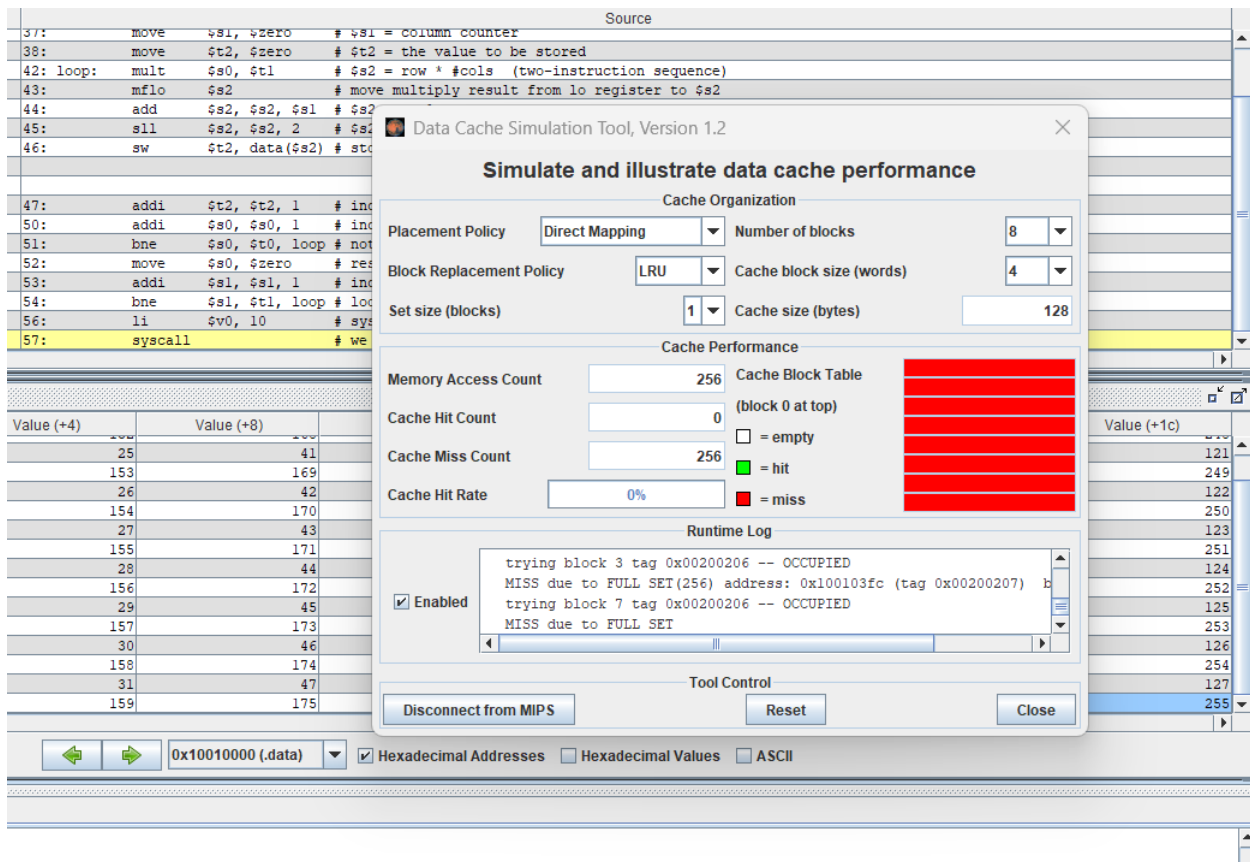
### Assignment 1:

-Khi chạy chương trình Fibonacci:



Hiệu suất của bộ nhớ đệm là 94%.

-Thực hiện 7 bước với chương trình column:



Khi chạy chương trình column, tỉ lệ cache hit rate là 0%

➔ Hiệu suất bộ nhớ đệm là 0%.

-Thực hiện chạy chương trình row:

Tỉ lệ cache hit rate là 75%. Vì với mỗi lần bỏ lỡ, một khối 4 words được ghi vào bộ đệm. Trong một đường truyền chính theo hàng, các phần tử của ma trận là được truy cập theo thứ tự chúng được lưu trữ trong bộ nhớ. Do đó, mỗi lần bỏ lỡ bộ đệm theo sau là 3 lần truy cập vì 3 phần tử tiếp theo được tìm thấy trong cùng một khối đệm. Tiếp theo là một lỗi khác khi ma trận trực tiếp ánh xạ tới khối bộ đệm tiếp theo và sau đó lặp lại chính nó. Vì vậy, 3 trong 4 lần truy cập bộ nhớ sẽ được giải quyết trong bộ nhớ 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 tỉ lệ cache hit rate còn 50%.

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

```

, $zero # $t2 = the value to be stored
, $t1 # $s2 = row * #cols (two-instruction sequence)
# move multiply result from lo register to $s2
, $s2, $s1 # $s2
, $s2, 2 # $s2
, data($s2) # store

```

**Data Cache Simulation Tool, Version 1.2**

### Simulate and illustrate data cache performance

**Cache Organization**

Placement Policy: **Direct Mapping** Number of blocks: **8**

Block Replacement Policy: **LRU** Cache block size (words): **4**

Set size (blocks): **1** Cache size (bytes): **128**

**Cache Performance**

Memory Access Count: **256** Cache Block Table (block 0 at top)

Cache Hit Count: **192** ☐ = empty

Cache Miss Count: **64** ☒ = hit

Cache Hit Rate: **75%** ☐ = miss

**Runtime Log**

☒ Enabled

```

(255) address: 0x100103f8 (tag 0x00200207) block range: 7-7
      trying block 7 tag 0x00200207 -- HIT
(256) address: 0x100103fc (tag 0x00200207) block range: 7-7
      trying block 7 tag 0x00200207 -- HIT

```

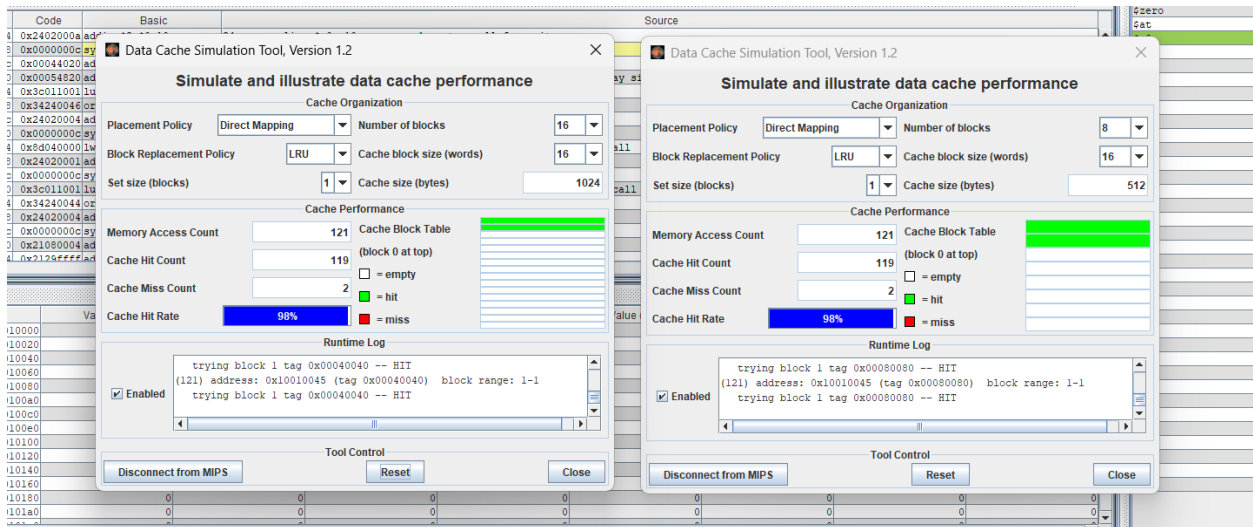
**Tool Control**

**Disconnect from MIPS** **Reset** **Close**

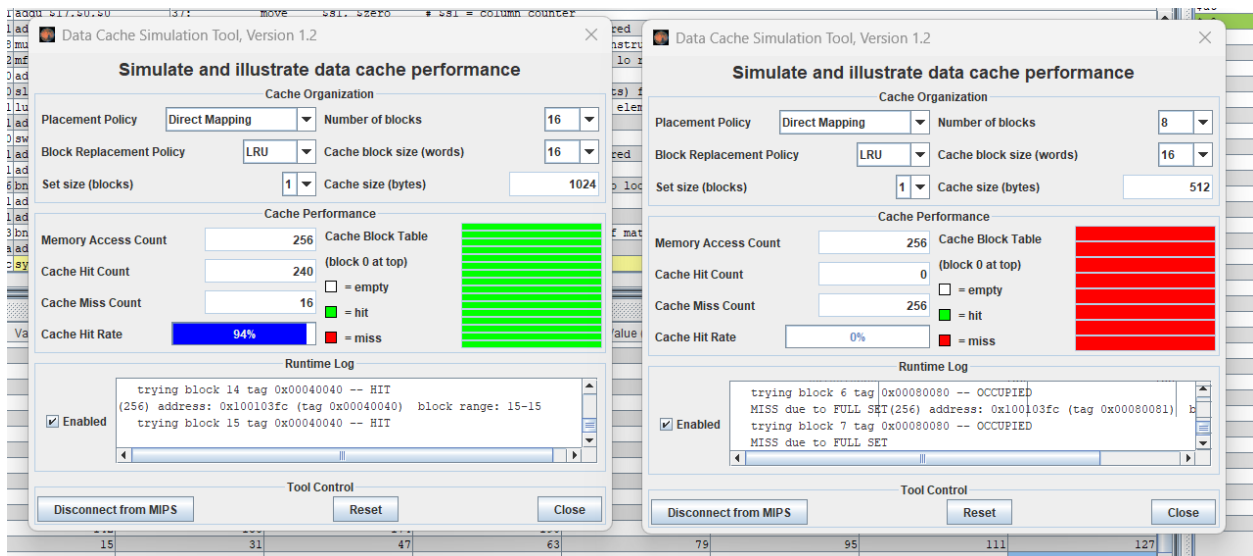
100 (data) ☒ Hexadecimal Addresses ☐ Hexadecimal Values ☐ ASCII

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

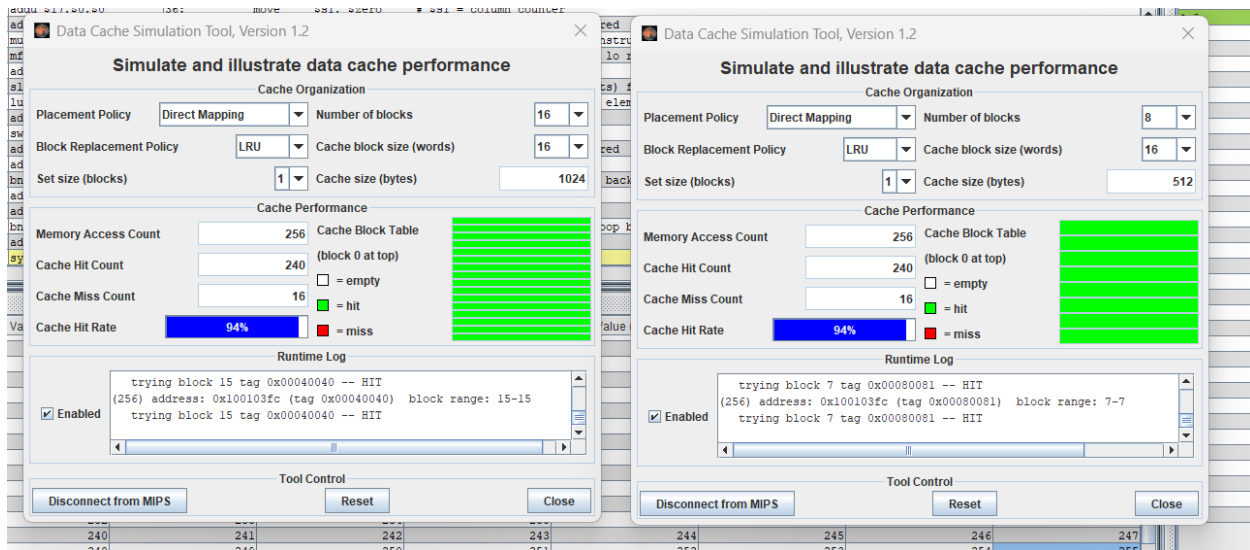
Khi chạy chương trình Fibonacci:



Khi chạy chương trình column:

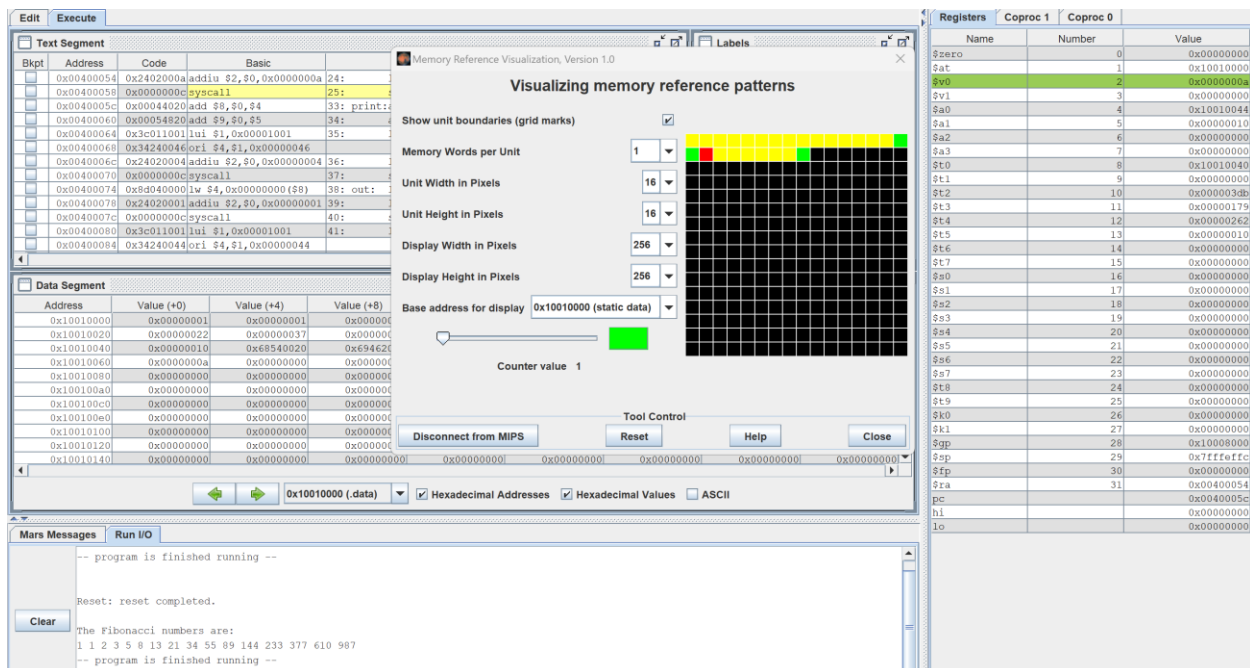


Khi chạy chương trình row:



## Assignment 2:

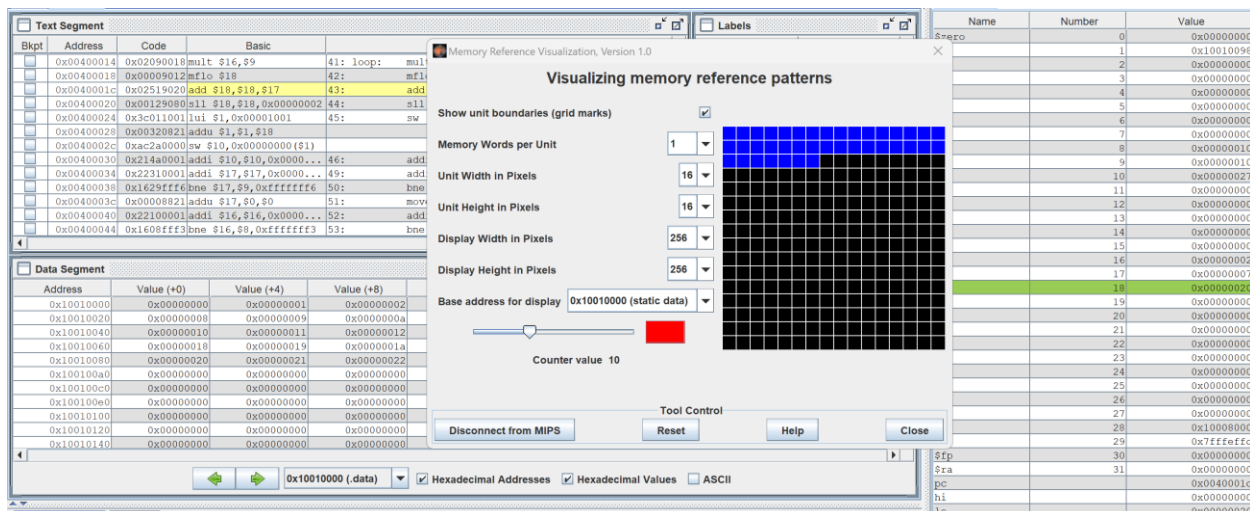
-Chương trình Fibonacci:



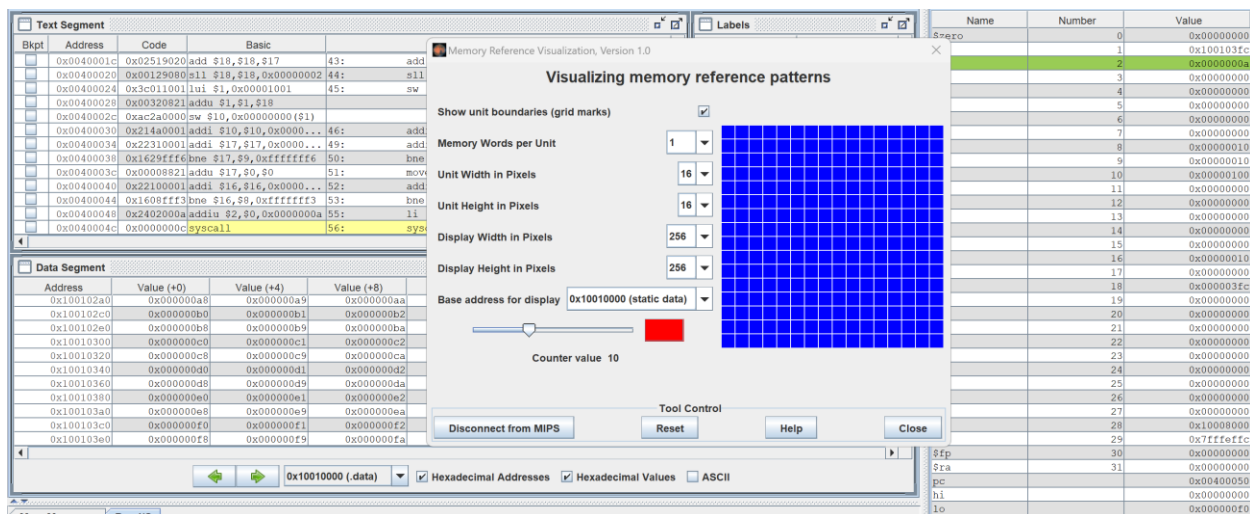
→ Đúng với lý thuyết.

-Chương trình row-major:

Khi dừng chương trình đột ngột:



Kết thúc chương trình:

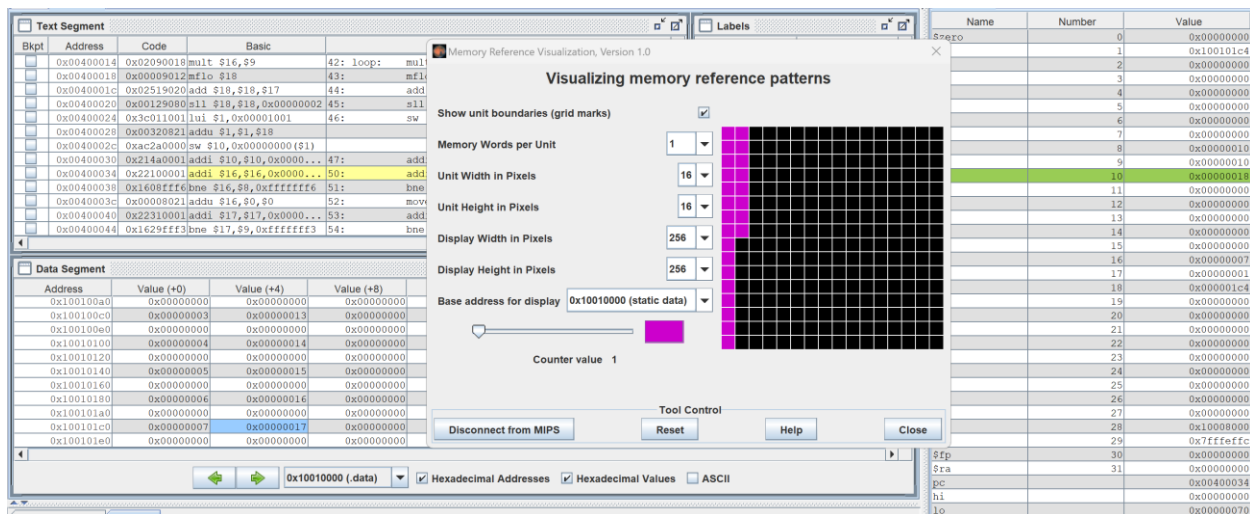


+Các phần tử được ghi theo hàng, từ trái qua phải, từ trên xuống dưới.

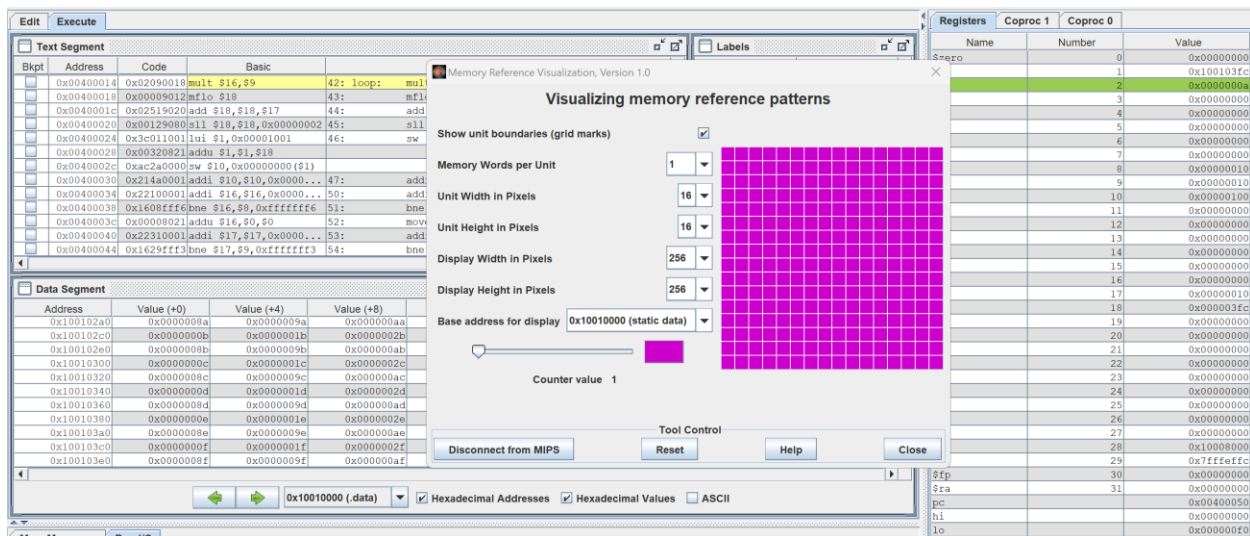
➔Đúng với lí thuyết.

-Chương trình column-major:

Khi dừng chương trình đột ngột:



Khi kết thúc chương trình:



+Các phần tử được ghi theo cột, từ trên xuống dưới, từ trái qua phải.

→Đúng với lí thuyết.