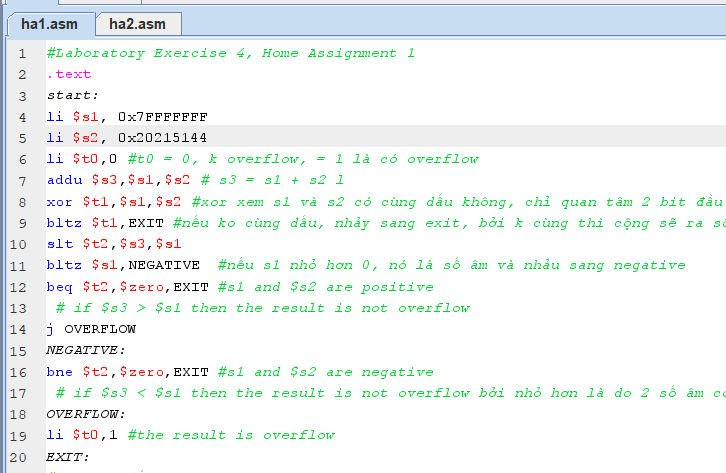
**Báo cáo thực hành**

**Họ và tên: Nguyễn Anh Thứ**

**MSSV: 20215144 (mssv được sử dụng cho 1 số bài)**

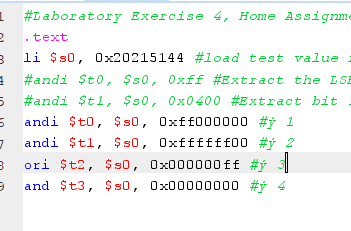
**Assignment 1:**



Các bước thu được khi chạy với 2 giá trị trên.

* Dòng 4, 5: nạp giá trị cho 2 thanh ghi $s1(0x7FFFFFFF ) và $s2 (0x20215144)
* Dòng 6: đặt t0 = 0, với mặc định 0 là không tràn số, 1 là tràn số
* Dòng 7: tính tổng s1, s2, lưu vào trong s3
* Dòng 8: xor s1 và s2, chỉ quan tâm đến bit dấu để xác định dấu,
* Dòng 9: do s1 và s2 cùng dấu, nên bỏ qua dòng này
* Dòng 11: s1 ở đây lớn hơn 0, nhảy xuống câu lệnh tiếp theo
* Dòng 10: so sánh s3 với s1, ở đây do thấy nhỏ hơn lên t2 = 1
* Dòng 12: ta thấy do t2= 1, chương trình tiếp tục chạt và nhảy đến Overflow

**Assignment 2:**



* Extract MSB of $s0: dùng andi với mask là 0xff000000
* Clear LSB of $s0: tương tự, ở đây là lấy hết bit trừ LSB, dùng andi với mask là 0xffffff00
* Set LSB of $s0 (bits 7 to 0 are set to 1): dùng ori với mask là 0x000000ff
* Clear $s0 (s0=0): tưởng tự ý 2, dùng andi với mask là 0x00000000

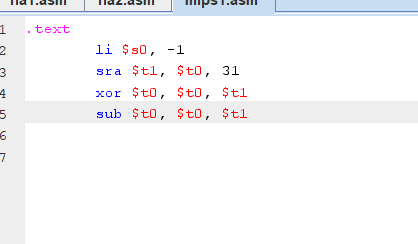


**Assignment 3:**

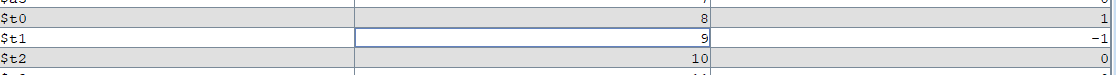
* abs $s0,$s1

$s0 <= |$s1|

Code:

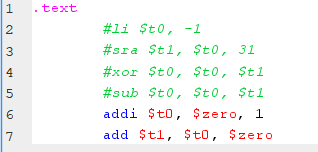


Kết quả:



* move $s0,$s1

$s0 <= $s1



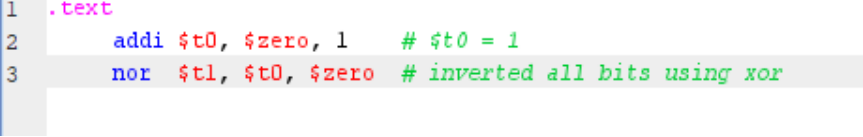
Kết quá:



* not $s0, $s1

$s0 <= bit invert ($s1)

Code:

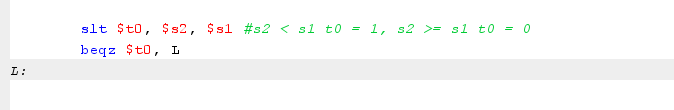


Kết quả:

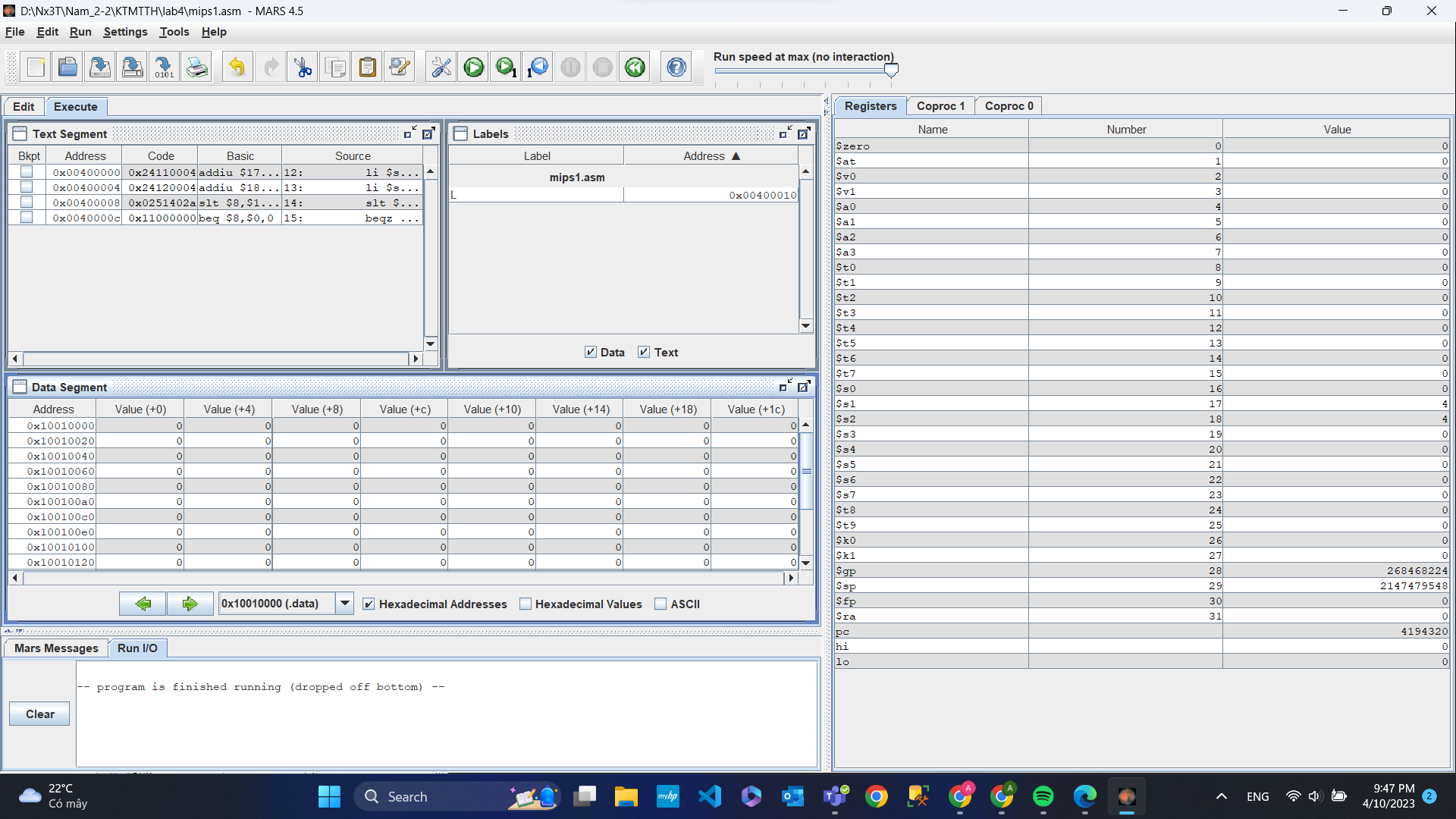


* ble $s1,s2,L

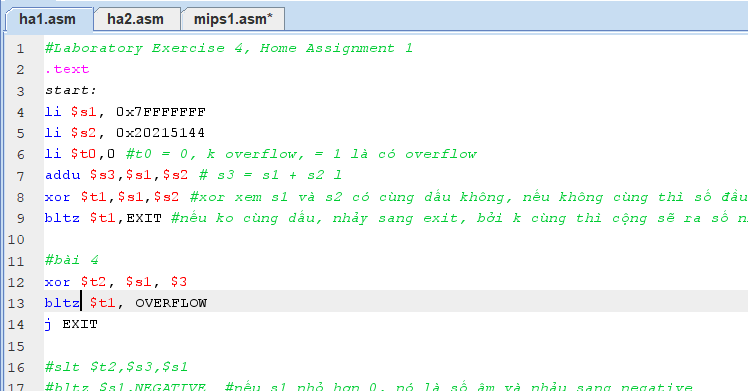
Code:



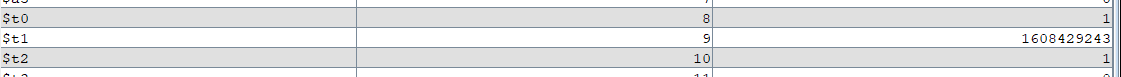
* Kết quả:



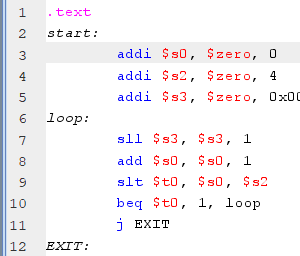
**Assignment 4:**



Kết quả khi chạy:



**Assignment 5:**



* Sử dụng lệnh sll, dịch trái 1 đơn vị mỗi vòng lặp, với mỗi kết quả s2, vòng lặp sẽ được thực hiện đến khi nào đến giá trị 2 mũ s2

