Laboratory Exercise 3

**Assignment 1:**



* *TH1*: Thiết lập giá trị i=5 , j=6



* Thanh ghi $s1 lưu giá trị của i, thanh ghi $s2 lưu giá trị của j. Nên khi thực hiện lệnh slt, do j > i nên $t0 = 0x00000000
* Khi chạy lệnh bne, $t0 = $zero nên sẽ chạy tuần tự các lệnh tiếp theo.
* Lệnh addi $t1,$t1,1 $t1 = 0x00000001
* Lệnh addi $t3, $zero, 1 $t3 = 0x00000001
* Lệnh j endif nhảy sang nhãn endif, bỏ qua nhãn else
* *TH2*: Thiết lập giá trị i=6, j=5



* j < i nên khi thực hiện lệnh slt, $t0 = 0x00000001
* Khi chạy lệnh bne $t0 khác $zero nên rẽ nhánh sang nhãn else
* Lệnh addi $t2, $t2,-1 $t2 = 0xffffffff
* Lệnh add $t3,$t3,$t3 $t3 = 0x00000000 (do z = 0)

**Assignment 2:**



* Thiết lập giá trị i = -1 , n = 3, mảng A{0,1,2} , step = 1, sum = 0.



* $s1 = 0xffffffff ; $s2 = 0x10010000 ; $s3 = 0x00000003 ;

$s4 = 0x00000001 ; $s5 = 0x00000000

* Vòng lặp đầu tiên
* Lệnh add $s1,$s1,$s4 gán i = i + step (i = -1 + 1 = 0) , sau lệnh này $s1 = 0x00000000
* Lệnh add $t1,$s1,$s1 và add $t1,$t1,$t1 dùng để lấy giá trị 4i, sau 2 lệnh này $t1 = 0x00000000
* Lệnh add $t1,$t1,$t2 dùng để lưu địa chỉ phần tử A[0] vào thanh ghi $t1, sau lệnh này $t1 = 0x10010000
* Lệnh lw $t0,0($t1) dùng để lưu giá trị của phần từ A[0] vào thanh ghi $t0, sau lệnh này $t0 = 0x00000000 do A[0] = 0
* Lệnh add $s5,$s5,$t0 gán sum = sum + A[0] , sau lệnh này $s5 = 0x00000000
* Lệnh bne $s1,$s3,loop : do i khác n = 3 nên quay lại nhãn loop thực hiện các vòng lặp tiếp theo
* Cứ thực hiện vòng lặp như vậy, ta được kết quả:
* $s1 = 0x00000003 (i =3)
* $s5 = 0x00000003 (sum = 3)
* Thực hiện với trường hợp: i = 0 , step = 2, n = 10, sum = 0, mảng A{0,1,2,3,4,5,6,7,8,9}



* Thực hiện tương tự như trường hợp đầu tiên, ta được kết quả:
* $s1 = 0x0000000a (i = 10)
* $s2 = 0x10010000 (A[0])
* $s3 = 0x0000000a (n = 10)
* $s4 = 0x00000002 (step =2)
* $s5 = 0x00000014 (sum = 20 = 2+4+6+8)

**Assignment 3:**





* Thực hiện các lệnh:
* Lệnh la $s0, test dùng để load địa chỉ biến test vào thanh shi $s0

$s0 = 0x10010000

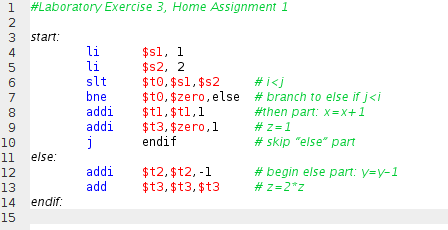
* Lệnh lw $s1,0($s0) load giá trị biến test vào thanh ghi $s1

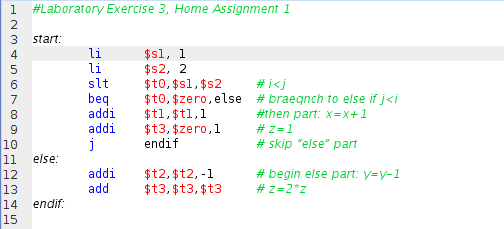
$s1 = 0x00000001

* Các lệnh li để gán giá trị cho các case: $t0 = 0x00000000 ; $t1 = 0x00000001 ; $t2 = 0x00000002
* Chạy lệnh beq $s1,$t0,case\_0 : do $s1 khác $t0 nên chạy tuần tự các lệnh tiếp theo
* Chạy lệnh beq $s1,$t1,case\_1 : do $s1 = $t1 nên rẽ nhánh đến nhãn case\_1
* Lệnh sub $s2,$s2,$t1 : $s2 = 0xffffffff
* Lệnh b continue : nhảy đến nhãn continue

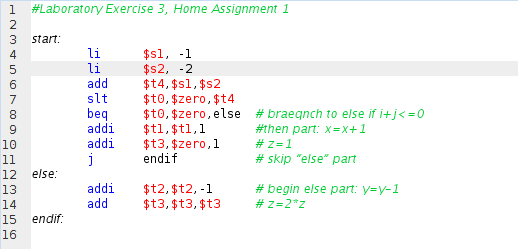
**Assignment 4:**

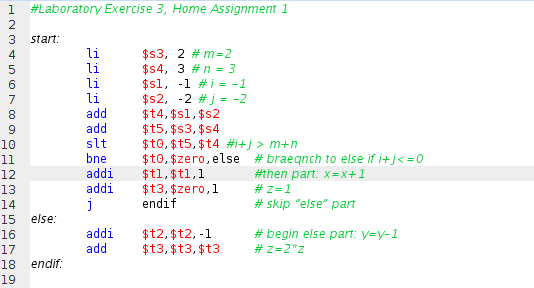


1. i < j
2. i >= j



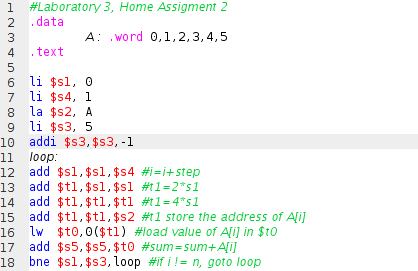
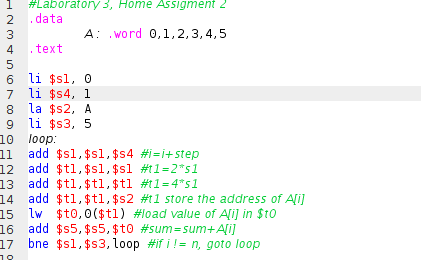
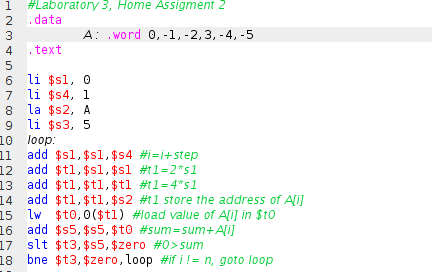
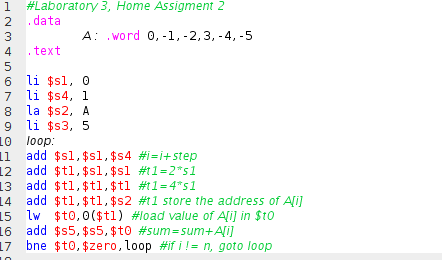
1. i+j <= 0



1. i+j > m+n

**Assignment 5:**

Modify the Assignment 2, so that the condition tested at the end of the loop is

1. i < n
2. i <= n
3. Sum >= 0
4. A[i] == 0

**Assignment 6:**



