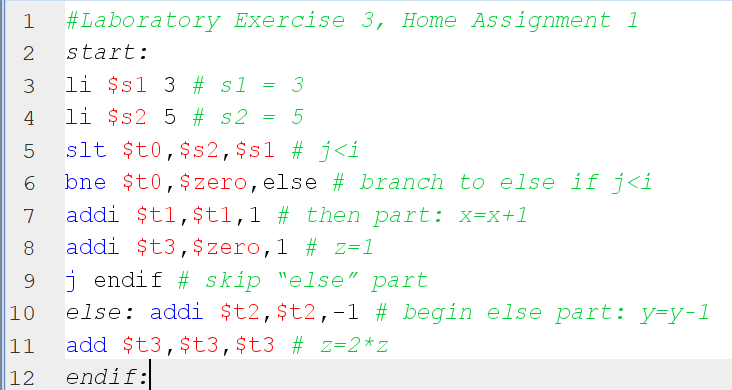
**Báo cáo thực hành bài 3**

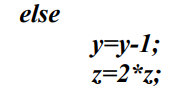
Họ và tên: Nguyễn Anh Thứ – MSSV: 20215144

Assignment 1:

Diagram

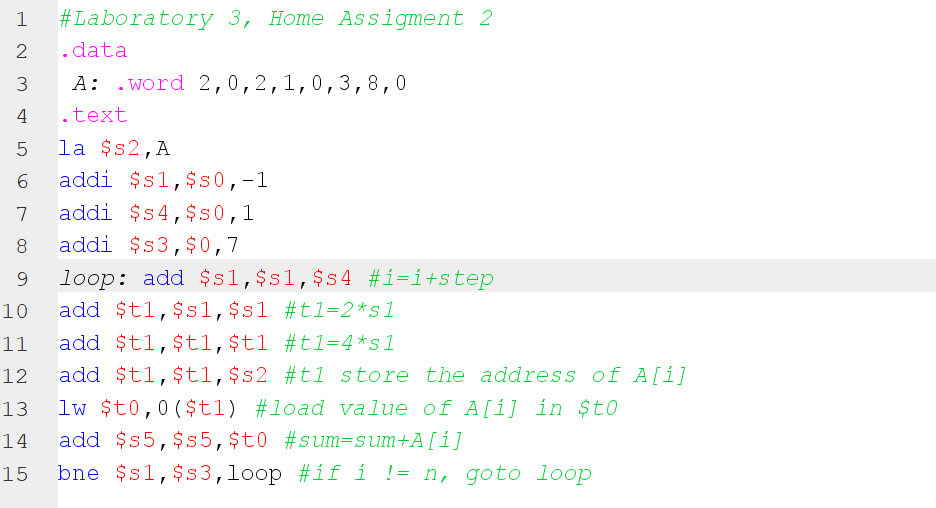
Description automatically generated

* li $s1 3 : khởi tạo giá trị 3 cho thanh ghi $s1
* li $s2 5 : khởi tạo giá trị 5 cho thanh ghi $s2
* slt $t0,$s2,$s1: So sánh nội dung của thanh ghi $1 và thanh ghi $2. Nếu giá trị của $s2 nhỏ hơn thanh ghi $1, kết quả của phép so sánh được ghi vào thanh ghi $t0 dưới dạng 1, ngược lại nếu giá trị của thanh ghi $s2 lớn hơn thanh ghi $2 thì kết quả được ghi vào thanh ghi $t0 dưới dạng 0
* bne $t0, $zero, else: Nếu giá trị trong thanh ghi $t0$ khác 0 (tức là $s2 < $s1), chương trình sẽ nhảy tới nhãn else để thực hiện các lệnh trong phần else. Nếu giá trị trong thanh ghi $t0$ bằng 0 (tức là $s2 > s1$), chương trình sẽ thực hiện tiếp các lệnh phía sau nhãn endif.
* addi $t1, $t1, 1 : Nếu giá trị trong thanh ghi $t0$ bằng 1 (tức là $s2 < $s1), chương trình sẽ thực hiện lệnh này, tăng giá trị của thanh ghi $t1$ lên 1.
* addi $t3, $zero, 1: Gán giá trị 1 vào thanh ghi $t3$.
* j endif: Nhảy tới endif và thực hiện các lệnh phía sau nhãn này.
* else: addi $t2, $t2, -1: Nếu giá trị trong thanh ghi $t0$ bằng 0 (tức là $s2 > $s1), chương trình sẽ thực hiện các lệnh trong phần else dưới đây. Lệnh này giảm giá trị trong thanh ghi $t2$ đi 1.



* add $t3, $t3, $t3: Nhân giá trị trong thanh ghi $t3$ với 2.
* endif: Kết thúc chương trình.

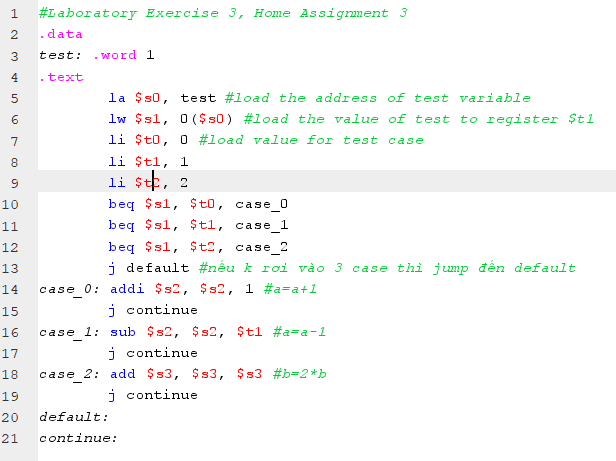
Home Assignment 2:

Graphical user interface, application, table

Description automatically generated

* Câu lệnh đầu tiên: Khai báo mảng A, nạp địa chỉ của mảng A vào thanh ghi $s2
* 
* Câu lệnh thứ 3 là câu lệnh khai báo chỉ số đầu tiên của mảng, sử dụng lệnh addi $s1,$s0,-1 để nạp giá trị -1 cho thanh ghi $s1 (mảng có 8 phần tử nên bắt đầu từ phần tử -1 và kết thúc ở phần tử thứ 7)
* 
* Câu lệnh thứ 4: khai báo step = 1 (bước nhảy của chỉ số), nạp vào thanh ghi $s4
* 
* Câu lệnh thứ 5: khai báo phần tử cuối cùng của mảng(7), nạp vào thanh ghi $s3
* Câu lệnh thứ 6,7 : thể hiện bước nhảy của địa chỉ thanh ghi (Khi bắt đầu vòng lặp, giá trị của chỉ số vòng lặp tăng dần theo step. Còn mỗi lần địa chỉ của mảng thay đổi sẽ nạp thêm 4 byte vào trong thanh ghi, mỗi lần thanh ghi t1 nạp thêm 2 byte nên sau 2 lần thì sẽ thành 4 byte
* Câu lệnh thứ 8: nạp đỉa chỉ của A[I] vào thanh ghi $t1
* Câu lệnh thứ 9: nạp giá trị A[i] vào thanh ghi $t0
* Câu lệnh thứ 10 nạp giá trị của tổng vào thanh ghi $s5 qua lệnh add $s5,$s5,$t0
* 
* Câu lệnh 11 rẽ nhánh, nếu chỉ số của mảng là giá trị của thanh ghi $s1 khác số phần tử của mảng ($s3) thì vòng lặp sẽ được tiếp tục
* Sau khi chạy vòng lặp đến I=8, vòng lặp sẽ kết thúc với kết quả (32bit) trên thanh ghi tổng $s5 là chuyển sang hệ thập phân là 15 (2+0+2+1+0+3+8+0=15)

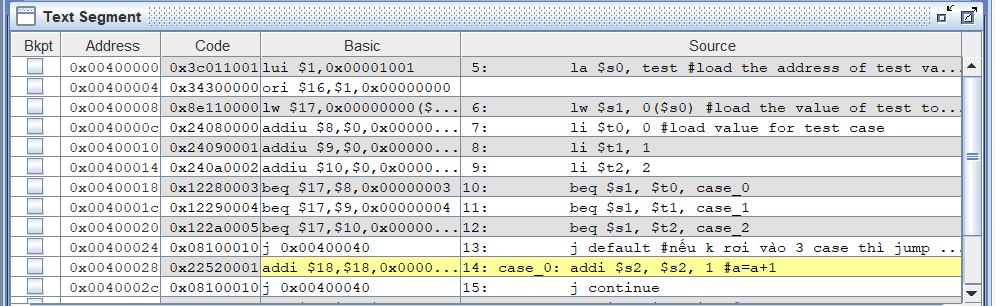
Assignment 3:



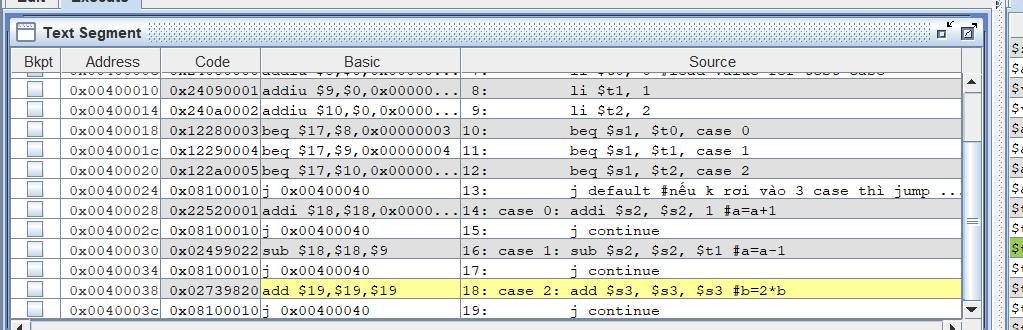
*Code*

* Dòng 3: lưu trữ giá trị của biến test là 1
* Dòng 5: lưu địa chỉ của test vào thanh ghi $s0
* Dòng 6: đọc giá trị của test từ địa chỉ đã được lưu trong $s0 vào $s1
* Dòng 7 – 9: tải các giá trị 0, 1, 2 lần lượt vào các thanh ghi $t0, $t1, $t2 để tạo giá trị củactest case
* Dòng 10 –12: dùng lệnh beq để so sánh giá trị các case với giá trị cần test. Nếu bằng nhau, chương trình nhảy đến case tương ứng. Nếu không có giá trị trùng khớp, chương trình nhảy đến nhãn default:
* Trong các case từ dòng 14, 16, 17: thực hiện các phép toán của đề bài, sau đó nhảy đến nhãn continue.

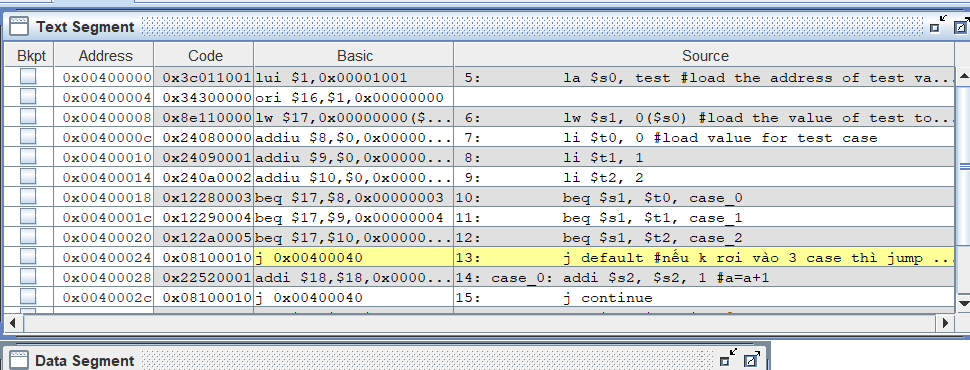
Khi thay đổi giá trị của test sang 0 hoặc 2, chương trình thực hiện các case\_0 và case\_2, sau đó nhảy đến nhãn continue. Với trường hợp test khác, chương trình nhảy đến default và kết thúc ở đó.



*Khi test = 0*



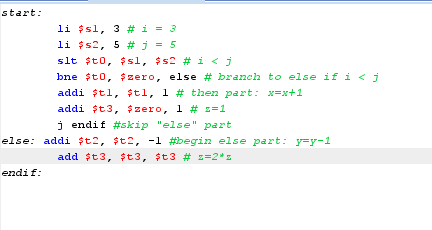
*Khi test = 2*

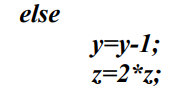
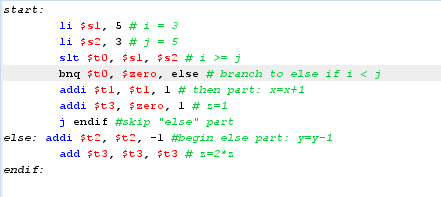
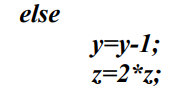


*Khi test = 4*

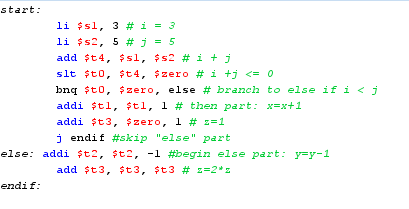
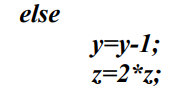
Assignment 4:

1. i<j

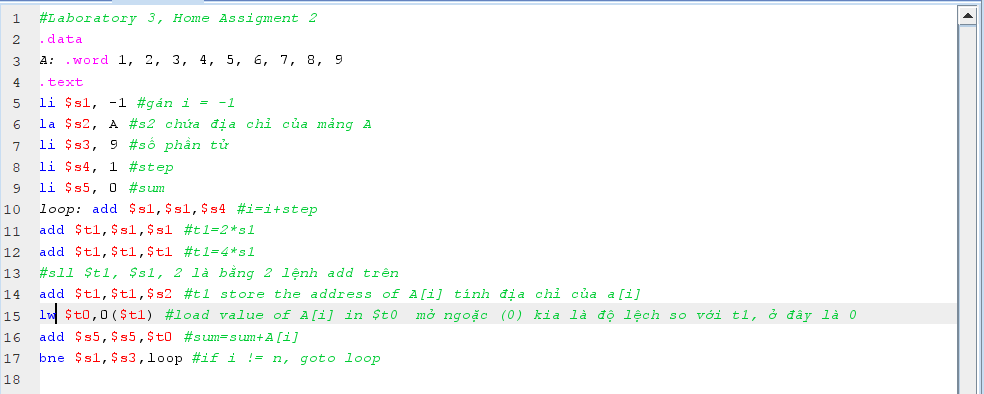


* li $s1 3 : khởi tạo giá trị 3 cho thanh ghi $s1
* $s2 5 : khởi tạo giá trị 5 cho thanh ghi $s2
* slt $t0,$s1,$s2: So sánh nội dung của thanh ghi $1 và thanh ghi $2. Nếu giá trị của $s1 nhỏ hơn thanh ghi $2, kết quả của phép so sánh được ghi vào thanh ghi $t1 dưới dạng 1, ngược lại nếu giá trị của thanh ghi $s1 lớn hơn thanh ghi $s2 thì kết quả được ghi vào thanh ghi $t1 dưới dạng 0
* bne $t0, $zero, else: Nếu giá trị trong thanh ghi $t0 khác 0 (tức là $s1 < $s2), chương trình sẽ nhảy tới nhãn else để thực hiện các lệnh trong phần else. Nếu giá trị trong thanh ghi $t0$ bằng 0 (tức là $s1 > $s2), chương trình sẽ thực hiện tiếp các lệnh phía sau nhãn endif.
* addi $t1, $t1, 1 : Nếu giá trị trong thanh ghi $t0$ bằng 1 (tức là $s2 < $s1), chương trình sẽ thực hiện lệnh này, tăng giá trị của thanh ghi $t1$ lên 1.
* addi $t3, $zero, 1: Gán giá trị 1 vào thanh ghi $t3$.
* j endif: Nhảy tới endif và thực hiện các lệnh phía sau nhãn này.
* else: addi $t2, $t2, -1: Nếu giá trị trong thanh ghi $t0$ bằng 0 (tức là $s2 > $s1), chương trình sẽ thực hiện các lệnh trong phần else dưới đây. Lệnh này giảm giá trị trong thanh ghi $t2$ đi 1.
* 
* add $t3, $t3, $t3: Nhân giá trị trong thanh ghi $t3$ với 2.
* endif: Kết thúc chương trình.
  + b. i>=j
* 
* li $s1 5 : khởi tạo giá trị 5 cho thanh ghi $s1
* li $s2 3 : khởi tạo giá trị 3 cho thanh ghi $s2
* slt $t0,$s2,$s1: So sánh nội dung của thanh ghi $1 và thanh ghi $2. Nếu giá trị của $s2 nhỏ hơn thanh ghi $1, kết quả của phép so sánh được ghi vào thanh ghi $t0 dưới dạng 1, ngược lại nếu giá trị của thanh ghi $s2 lớn hơn thanh ghi $2 thì kết quả được ghi vào thanh ghi $to dưới dạng 0
* bnq $t0, $zero, else: Nếu giá trị trong thanh ghi $t0 khác 0 (tức là $s1 >= $s2), chương trình sẽ nhảy tới nhãn else để thực hiện các lệnh trong phần else. Nếu giá trị trong thanh ghi $t0$ bằng 0 (tức là $s1 < $s2), chương trình sẽ thực hiện tiếp các lệnh phía sau nhãn endif.
* addi $t1, $t1, 1 : Nếu giá trị trong thanh ghi $t0$ bằng 1 (tức là $s2 < $s1), chương trình sẽ thực hiện lệnh này, tăng giá trị của thanh ghi $t1$ lên 1.
* addi $t3, $zero, 1: Gán giá trị 1 vào thanh ghi $t3$.
* j endif: Nhảy tới endif và thực hiện các lệnh phía sau nhãn này.
* else: addi $t2, $t2, -1: Nếu giá trị trong thanh ghi $t0$ bằng 0 (tức là $s2 > $s1), chương trình sẽ thực hiện các lệnh trong phần else dưới đây. Lệnh này giảm giá trị trong thanh ghi $t2$ đi 1.
* 
* dd $t3, $t3, $t3: Nhân giá trị trong thanh ghi $t3$ với 2.
* endif: Kết thúc chương trình.

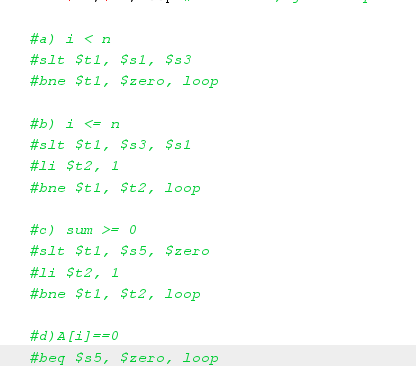
c. i+j <= 0

* 
* li $s1 3 : khởi tạo giá trị 3 cho thanh ghi $s1
* li $s2 5 : khởi tạo giá trị 5 cho thanh ghi $s2
* add $t4, $s1, $s2 : ghi giá trị của tổng thanh ghi $s1 và $s2
* slt $t0,$t4,$zero: So sánh nội dung của thanh ghi $t4 và thanh ghi $zero. Nếu giá trị của $t4 nhỏ hơn thanh ghi $zero, kết quả của phép so sánh được ghi vào thanh ghi $t1 dưới dạng 1, ngược lại nếu giá trị của thanh ghi $s2 lớn hơn thanh ghi $2 thì kết quả được ghi vào thanh ghi $t1 dưới dạng 0
* bnq $t0, $zero, else: Nếu giá trị trong thanh ghi $t0 khác 0 (tức là $t4 <= 0), chương trình sẽ nhảy tới nhãn else để thực hiện các lệnh trong phần else. Nếu giá trị trong thanh ghi $t0 bằng 0 (tức là $t4 > 0), chương trình sẽ thực hiện tiếp các lệnh phía sau nhãn endif.
* addi $t1, $t1, 1 : Nếu giá trị trong thanh ghi $t0$ bằng 1 (tức là $s2 < $s1), chương trình sẽ thực hiện lệnh này, tăng giá trị của thanh ghi $t1$ lên 1.
* addi $t3, $zero, 1: Gán giá trị 1 vào thanh ghi $t3$.
* j endif: Nhảy tới endif và thực hiện các lệnh phía sau nhãn này.
* else: addi $t2, $t2, -1: Nếu giá trị trong thanh ghi $t0$ bằng 0 (tức là $s2 > $s1), chương trình sẽ thực hiện các lệnh trong phần else dưới đây. Lệnh này giảm giá trị trong thanh ghi $t2$ đi 1.
* 
* add $t3, $t3, $t3: Nhân giá trị trong thanh ghi $t3$ với 2.
* endif: Kết thúc chương trình.

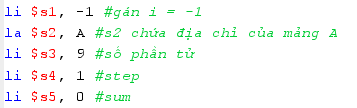
Assignment 5:



*Code mẫu.*



*Điều kiện đã sửa đổi của các ý*



*Thanh ghi và biến tương ứng*

a)

* Dùng slt: nếu s1<s3, t1 = 1.
* bne: nếu t1 khác 0, nhảy đến nhãn loop

b)

* Slt : nếu s3 < s1, t1 = 1
* Li: lưu t2 = 1
* Bne: nếu t1 khác t2, nghĩa là t1 = 0, nhảy đến nhãn loop

c)

* Slt: nếu s5 < 0, t1 = 1
* Li: lưu t2 = 1
* Bne nếu t1 khác t2, nghĩa là t1 = 0, s5 >= 0, nhảy đến nhãn loop

d)

* beq: nếu s5 = 0, nhảy đến nhãn loop.