**Báo cáo Thực hành KTMT buổi 6**

**Họ và tên:** Nguyễn Đức Phú

**MSSV:** 20215116

**Assignment 1: Maximum-sum prefix**

* Bổ sung đoạn code nhập dữ liệu:

.data

A: .word

message1: .asciiz "Nhap so phan tu: "

message2: .asciiz "Nhap lan luot cac phan tu:\n"

.text

main:

li $v0, 4

la $a0, message1

syscall

li $v0, 5

syscall

add $a1,$zero,$v0 # $a1 chua so phan tu

li $v0, 4

la $a0, message2

syscall

la $a0,A

addi $t0,$zero,0

input:

beq $t0,$a1,end\_input

add $t2,$t0,$t0

add $t2,$t2,$t2

add $t3,$t2,$a0 # address of A[i] in $t3

li $v0, 5

syscall

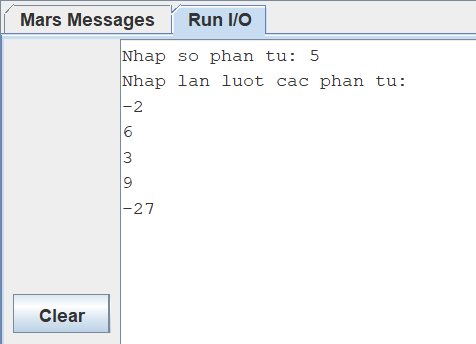
sw $v0,0($t3)

addi $t0,$t0,1

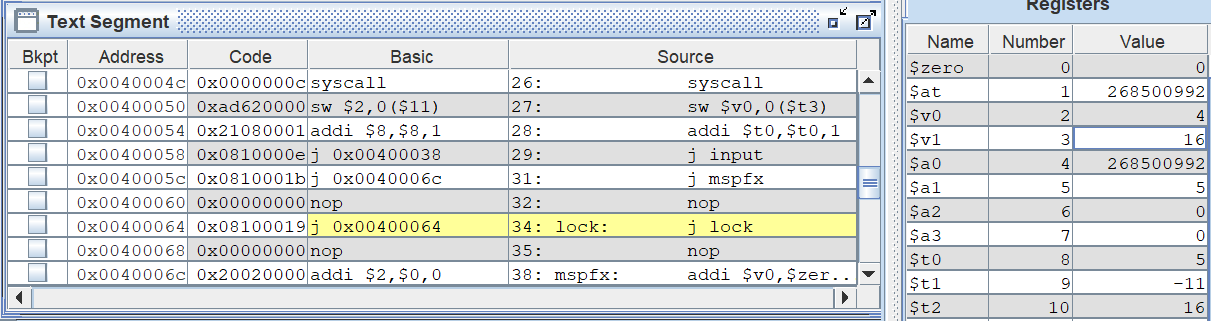
j input

end\_input:

* Với bộ dữ liệu đầu vào như sau:



* Kết quả cho ra tại thanh ghi $v1 = 16 và chỉ số tại $v0 = 4 là kết quả đúng



**Assignment 2: Selection Sort**

* Code:

.data

A: .word -2,5,7,-23,45,-6,34,2

Aend: .word

message1: .asciiz " "

message2: .asciiz "\n"

.text

main:

la $a0,A #$a0 = Address(A[0])

la $a1,Aend

la $t7,Aend #Use in print array

addi $a1,$a1,-4 #$a1 = Address(A[n-1])

j sort #sort

after\_sort:

li $v0, 10 #exit

syscall

end\_main:

sort:

beq $a0,$a1,done #list is sorted

j max #call the max procedure

after\_max:

lw $t0,0($a1) #load last element into $t0

sw $t0,0($v0) #copy last element to max location

sw $v1,0($a1) #copy max value to last element

addi $a1,$a1,-4 #decrement pointer to last element

print\_arr:

la $t6,A

j show\_arr

end\_print:

j sort #repeat sort for smaller list

done:

j after\_sort

max:

addi $v0,$a0,0 #init max pointer to first element

lw $v1,0($v0) #init max value to first value

addi $t0,$a0,0 #init next pointer to first

loop:

beq $t0,$a1,ret #if next=last, return

addi $t0,$t0,4 #advance to next element

lw $t1,0($t0) #load next element into $t1

slt $t2,$t1,$v1 #(next)<(max) ?

bne $t2,$zero,loop #if (next)<(max), repeat

addi $v0,$t0,0 #next element is new max element

addi $v1,$t1,0 #next value is new max value

j loop #change completed; now repeat

ret:

j after\_max

show\_arr:

li $v0,1

lw $a0,0($t6)

syscall

li $v0, 4

la $a0, message1

syscall

addi $t6,$t6,4

bne $t6,$t7,show\_arr

li $v0, 4

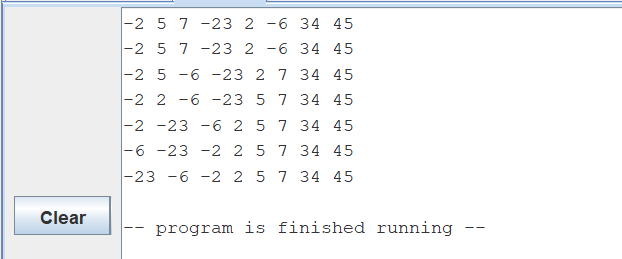
la $a0, message2

syscall

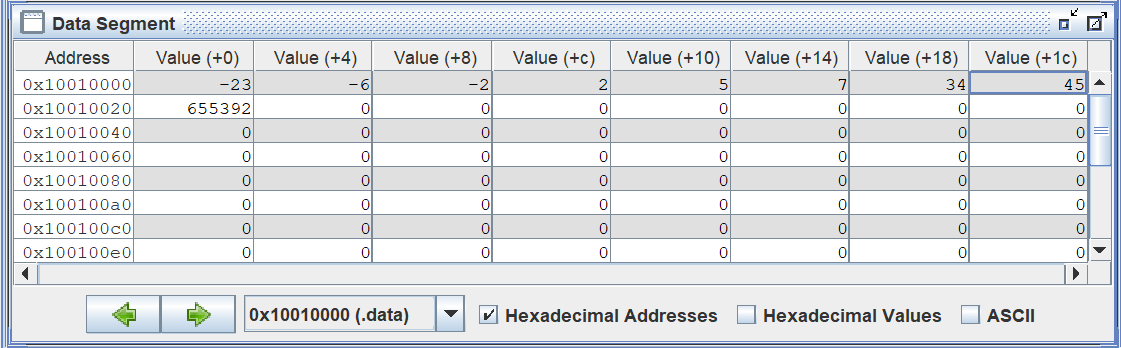
la $a0,A

j end\_print

* Kết quả:
* In ra array sau mỗi lần đổi chỗ:



* Kết quả tại Data Segment:



**Assignment 3: Bubble Sort**

* Code:

.data

A: .word 6,9,-3,27,20,-5,-12,8

Aend: .word

message1: .asciiz " "

message2: .asciiz "\n"

.text

la $a0, A

la $a1, Aend

la $t7, Aend

li $s0, 0 # count = 0 (count la bien dem phan tu)

li $s1, -1 # i = -1 (i trong loopi)

DemPhanTu:

beq $a1, $a0, Size

addi $a1, $a1, -4

addi $s0, $s0, 1

j DemPhanTu

Size:

addi $t0, $s0, -1 # t0 = So luong phan tu mang A - 1

loop1:

addi $s1, $s1, 1 # i++

li $s2, 0 # j = 0 (j trong loop2)

beq $s1, $t0, Exit # Neu i = size - 1 thì thoát

loop2:

sub $t2, $t0, $s1 # t2 = (size - 1) - i

beq $s2, $t2, loop1 # Neu j = (size - 1) - i thì nhay loop1

if\_swap:

sll $t3, $s2, 2

add $s3, $a0, $t3

lw $v0, 0($s3)

addi $s3, $s3, 4

lw $v1, 0($s3)

sle $t4, $v0, $v1

beq $t4, $zero, swap

addi $s2, $s2, 1

j loop2

swap:

sw $v0, 0($s3)

addi $s3, $s3, -4

sw $v1, 0($s3)

addi $s2, $s2, 1

print\_arr:

la $t6,A

j show\_arr

end\_print:

j loop2

show\_arr:

li $v0,1

lw $a0,0($t6)

syscall

li $v0, 4

la $a0, message1

syscall

addi $t6,$t6,4

bne $t6,$t7,show\_arr

li $v0, 4

la $a0, message2

syscall

la $a0,A

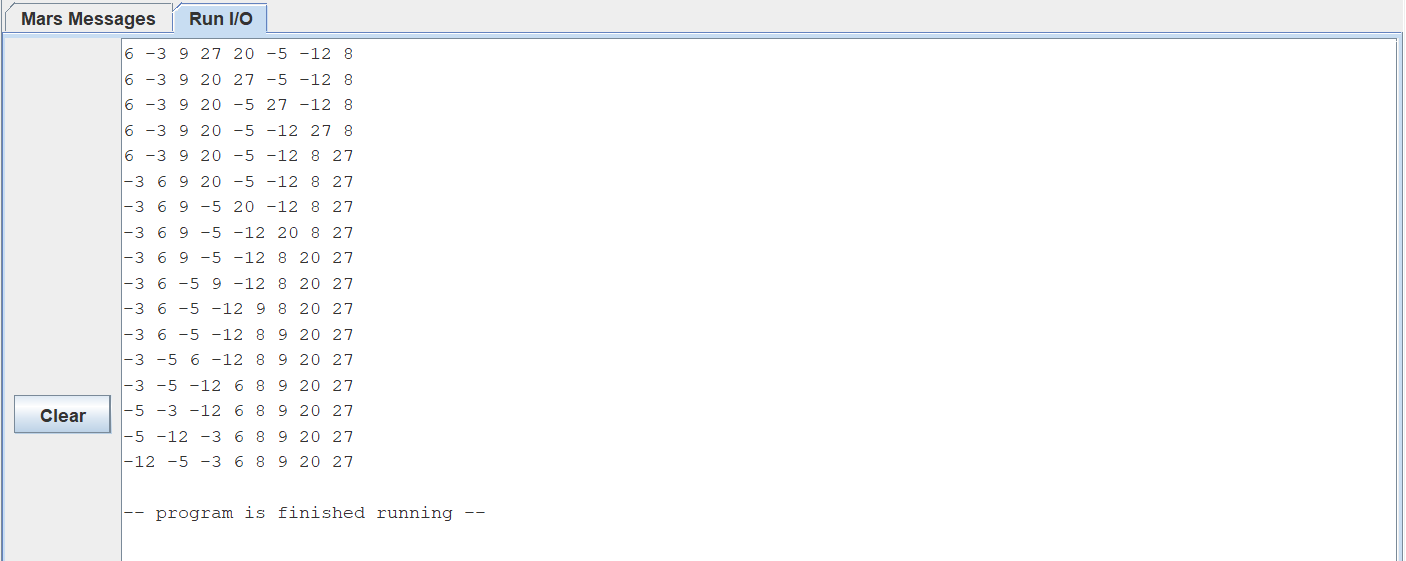
j end\_print

Exit:

li $v0, 10

syscall

* Kết quả chạy:
* Hiển thị array sau mỗi lần swap:



* Kết quả tại Data Segment:

Table

Description automatically generated with medium confidence

**Assignment 4: Insertion Sort**

* Code:

.data

A: .word 6,9,-3,27,20,-5,-12,8

Aend: .word

message1: .asciiz " "

message2: .asciiz "\n"

.text

la $a0, A

la $a1, Aend

la $t7, Aend #Use to Print Array

li $s0, 0 # count = 0 (dem phan tu)

li $s1, 0 # key = 0

li $s2, 0 # j = 0

li $s3, 1 # i = 1

DemPhanTu:

beq $a1, $a0, Loop

addi $a1, $a1, -4

addi $s0, $s0, 1

j DemPhanTu

Loop:

beq $s3, $s0, Exit

sll $t0, $s3, 2

add $s4, $a0, $t0

lw $s1, 0($s4)

addi $s2, $s3, -1

While:

slt $t1, $s2, $zero

sll $t0, $s2, 2

add $s5, $a0, $t0

lw $t3, 0($s5)

sle $t4, $t3, $s1

add $t1, $t1, $t4

bne $t1, $zero, loop\_continue

addi $s5, $s5, 4

sw $t3, 0($s5)

addi $s2, $s2, -1

j While

loop\_continue:

addi $s5, $s5, 4

sw $s1, 0($s5)

addi $s3, $s3, 1

print\_arr:

la $t6,A

j show\_arr

end\_print:

j Loop

show\_arr:

li $v0,1

lw $a0,0($t6)

syscall

li $v0, 4

la $a0, message1

syscall

addi $t6,$t6,4

bne $t6,$t7,show\_arr

li $v0, 4

la $a0, message2

syscall

la $a0,A

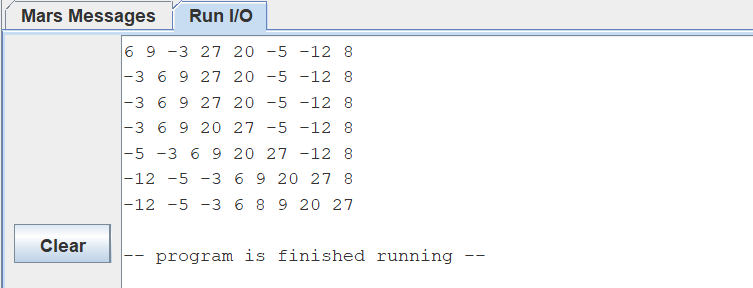
j end\_print

Exit:

li $v0, 10

syscall

* Kết quả chạy:
* Hiển thị Array sau mỗi vòng:



* Kết quả tại cửa sổ Data Segment:

