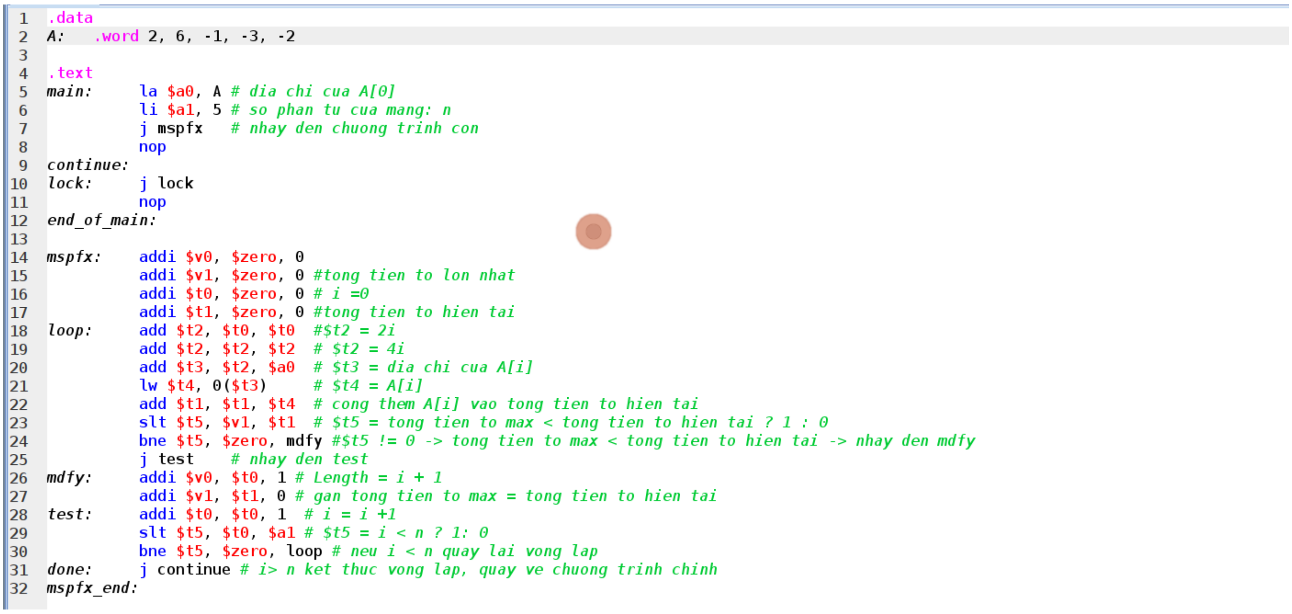
BÁO CÁO THỰC HÀNH - TUẦN 6
Lương Thị Tâm – 20194663

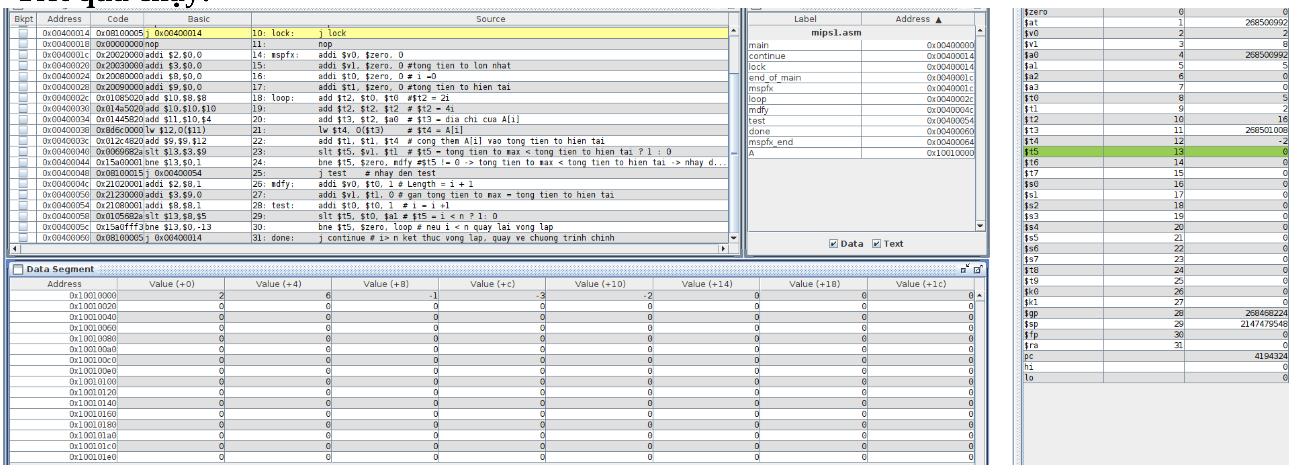
Bài 1:

*Code:



```
.data
A:    .word 2, 6, -1, -3, -2

.text
main:       la $a0, A # dia chi cua A[0]
            li $a1, 5 # so phan tu cua mang: n
            j mspfx   # nhay den chuong trinh con
            nop
continue:
lock:       j lock
            nop
end_of_main:

mspfx:      addi $v0, $zero, 0
            addi $v1, $zero, 0 #tong tien to lon nhat
            addi $t0, $zero, 0 # i =0
            addi $t1, $zero, 0 #tong tien to hien tai
loop:       add $t2, $t0, $t0  #$t2 = 2i
            add $t2, $t2, $t2  # $t2 = 4i
            add $t3, $t2, $a0  # $t3 = dia chi cua A[i]
            lw $t4, 0($t3)     # $t4 = A[i]
            add $t1, $t1, $t4  # cong them A[i] vao tong tien to hien tai
            slt $t5, $v1, $t1  # $t5 = tong tien to max < tong tien to hien tai ? 1 : 0
            bne $t5, $zero, mdfy #$t5 != 0 -> tong tien to max < tong tien to hien tai -> nhay den mdfy
            j test    # nhay den test
mdfy:       addi $v0, $t0, 1 # Length = i + 1
            addi $v1, $t1, 0 # gan tong tien to max = tong tien to hien tai
test:       addi $t0, $t0, 1  # i = i +1
            slt $t5, $t0, $a1 # $t5 = i < n ? 1: 0
            bne $t5, $zero, loop # neu i < n quay lai vong lap
done:       j continue # i> n ket thuc vong lap, quay ve chuong trinh chinh
mspfx_end:
```

*Kết quả chạy:



Bkpt	Address	Code	Basic	Source
□	0x00400014	0x08100005	j 0x00400014	10: lock: j lock
□	0x00400018	0x00000000	nop	11: nop
□	0x0040001c	0x20020000	addi $2,$0,0	14: mspfx: addi $v0, $zero, 0
□	0x00400020	0x20030000	addi $3,$0,0	15: addi $v1, $zero, 0 #tong tien to lon nhat
□	0x00400024	0x20080000	addi $8,$0,0	16: addi $t0, $zero, 0 # i =0
□	0x00400028	0x20090000	addi $9,$0,0	17: addi $t1, $zero, 0 #tong tien to hien tai
□	0x0040002c	0x01085020	add $10,$8,$8	18: loop: add $t2, $t0, $t0 #$t2 = 2i
□	0x00400030	0x014a5020	add $10,$10,$10	19: add $t2, $t2, $t2 # $t2 = 4i
□	0x00400034	0x01445820	add $11,$10,$4	20: add $t3, $t2, $a0 # $t3 = dia chi cua A[i]
□	0x00400038	0x8d6c0000	lw $12,0($11)	21: lw $t4, 0($t3) # $t4 = A[i]
□	0x0040003c	0x012c4820	add $9,$9,$12	22: add $t1, $t1, $t4 # cong them A[i] vao tong tien to hien tai
□	0x00400040	0x0069682a	slt $13,$3,$9	23: slt $t5, $v1, $t1 # $t5 = tong tien to max < tong tien to hien tai ? 1 : 0
□	0x00400044	0x15a00001	bne $13,$0,1	24: bne $t5, $zero, mdfy #$t5 != 0 -> tong tien to max < tong tien to hien tai -> nhay d...
□	0x00400048	0x08100015	j 0x00400054	25: j test # nhay den test
□	0x0040004c	0x21020001	addi $2,$8,1	26: mdfy: addi $v0, $t0, 1 # Length = i + 1
□	0x00400050	0x21230000	addi $3,$9,0	27: addi $v1, $t1, 0 # gan tong tien to max = tong tien to hien tai
□	0x00400054	0x21080001	addi $8,$8,1	28: test: addi $t0, $t0, 1 # i = i +1
□	0x00400058	0x0105682a	slt $13,$8,$5	29: slt $t5, $t0, $a1 # $t5 = i < n ? 1: 0
□	0x0040005c	0x15a0fff3	bne $13,$0,-13	30: bne $t5, $zero, loop # neu i < n quay lai vong lap
□	0x00400060	0x08100005	j 0x00400014	31: done: j continue # i> n ket thuc vong lap, quay ve chuong trinh chinh

Label	Address ▲
mips1.asm	
main	0x00400000
continue	0x00400014
lock	0x00400014
end_of_main	0x0040001c
mspfx	0x0040001c
loop	0x0040002c
mdfy	0x0040004c
test	0x00400054
done	0x00400060
mspfx_end	0x00400064
A	0x10010000

☑ Data ☑ Text

$zero	0	0
$at	1	268500992
$v0	2	2
$v1	3	8
$a0	4	268500992
$a1	5	5
$a2	6	0
$a3	7	0
$t0	8	5
$t1	9	2
$t2	10	16
$t3	11	268501008
$t4	12	-2
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	0
$s2	18	0
$s3	19	0
$s4	20	0
$s5	21	0
$s6	22	0
$s7	23	0
$t8	24	0
$t9	25	0
$k0	26	0
$k1	27	0
$gp	28	268468224
$sp	29	2147479548
$fp	30	0
$ra	31	0
pc		4194324
hi		0
lo		0

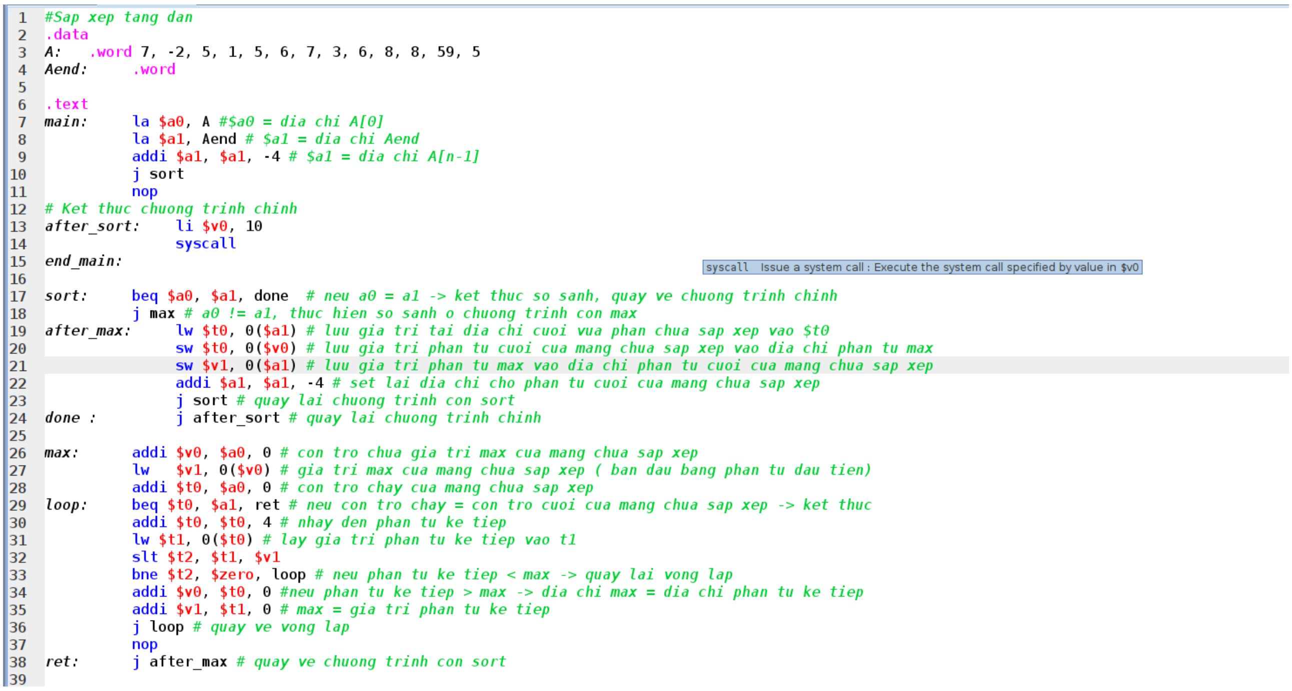
Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	2	6	-1	-3	-2	0	0	0
0x10010020	0	0	0	0	0	0	0	0
0x10010040	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0
0x100100e0	0	0	0	0	0	0	0	0
0x10010100	0	0	0	0	0	0	0	0
0x10010120	0	0	0	0	0	0	0	0
0x10010140	0	0	0	0	0	0	0	0
0x10010160	0	0	0	0	0	0	0	0
0x10010180	0	0	0	0	0	0	0	0
0x100101a0	0	0	0	0	0	0	0	0
0x100101c0	0	0	0	0	0	0	0	0
0x100101e0	0	0	0	0	0	0	0	0

Bài 2:

a, Sắp xếp tăng dần:

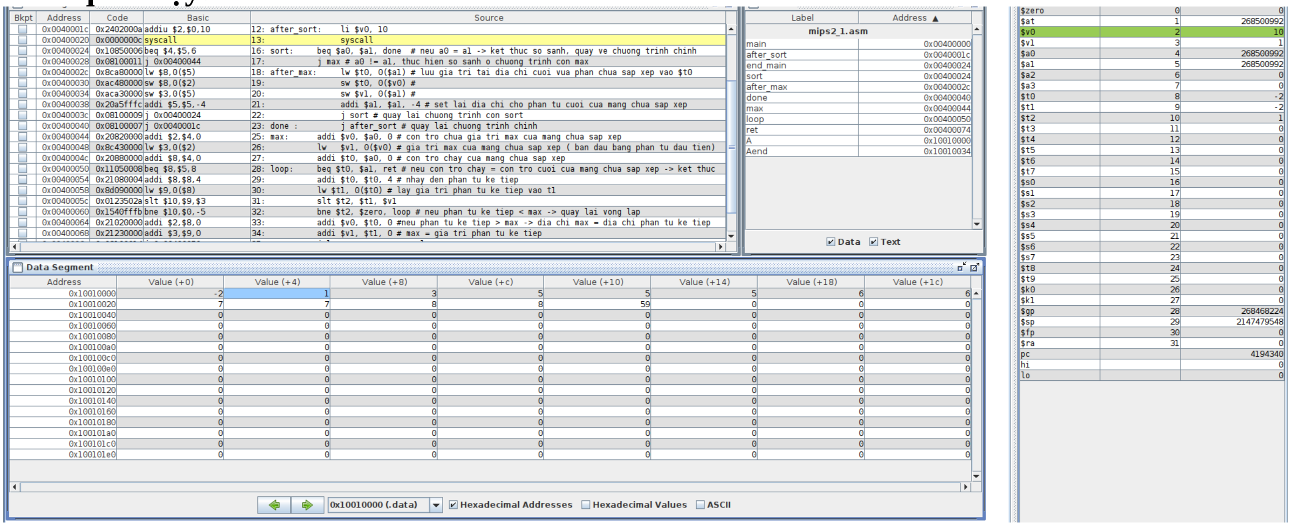
*Code:



```
#Sap xep tang dan
.data
A:    .word 7, -2, 5, 1, 5, 6, 7, 3, 6, 8, 8, 59, 5
Aend:      .word

.text
main:      la $a0, A #$a0 = dia chi A[0]
           la $a1, Aend # $a1 = dia chi Aend
           addi $a1, $a1, -4 # $a1 = dia chi A[n-1]
           j sort
           nop
# Ket thuc chuong trinh chinh
after_sort:     li $v0, 10
                syscall
end_main:

sort:      beq $a0, $a1, done  # neu a0 = a1 -> ket thuc so sanh, quay ve chuong trinh chinh
           j max # a0 != a1, thuc hien so sanh o chuong trinh con max
after_max:      lw $t0, 0($a1) # luu gia tri tai dia chi cuoi vua phan chua sap xep vao $t0
                sw $t0, 0($v0) # luu gia tri phan tu cuoi cua mang chua sap xep vao dia chi phan tu max
                sw $v1, 0($a1) # luu gia tri phan tu max vao dia chi phan tu cuoi cua mang chua sap xep
                addi $a1, $a1, -4 # set lai dia chi cho phan tu cuoi cua mang chua sap xep
                j sort # quay lai chuong trinh con sort
done :          j after_sort # quay lai chuong trinh chinh

max:       addi $v0, $a0, 0 # con tro chua gia tri max cua mang chua sap xep
           lw   $v1, 0($v0) # gia tri max cua mang chua sap xep ( ban dau bang phan tu dau tien)
           addi $t0, $a0, 0 # con tro chay cua mang chua sap xep
loop:      beq $t0, $a1, ret # neu con tro chay = con tro cuoi cua mang chua sap xep -> ket thuc
           addi $t0, $t0, 4 # nhay den phan tu ke tiep
           lw $t1, 0($t0) # lay gia tri phan tu ke tiep vao t1
           slt $t2, $t1, $v1
           bne $t2, $zero, loop # neu phan tu ke tiep < max -> quay lai vong lap
           addi $v0, $t0, 0 #neu phan tu ke tiep > max -> dia chi max = dia chi phan tu ke tiep
           addi $v1, $t1, 0 # max = gia tri phan tu ke tiep
           j loop # quay ve vong lap
           nop
ret:       j after_max # quay ve chuong trinh con sort
```

syscall Issue a system call : Execute the system call specified by value in $v0

*Kết quả chạy:



Bkpt	Address	Code	Basic	Source
□	0x0040001c	0x2402000a	addiu $2,$0,10	12: after_sort: li $v0, 10
□	0x00400020	0x0000000c	syscall	13: syscall
□	0x00400024	0x10850006	beq $4,$5,6	16: sort: beq $a0, $a1, done # neu a0 = a1 -> ket thuc so sanh, quay ve chuong trinh chinh
□	0x00400028	0x08100011	j 0x00400044	17: j max # a0 != a1, thuc hien so sanh o chuong trinh con max
□	0x0040002c	0x8ca80000	lw $8,0($5)	18: after_max: lw $t0, 0($a1) # luu gia tri tai dia chi cuoi vua phan chua sap xep vao $t0
□	0x00400030	0xac480000	sw $8,0($2)	19: sw $t0, 0($v0) #
□	0x00400034	0xaca30000	sw $3,0($5)	20: sw $v1, 0($a1) #
□	0x00400038	0x20a5fffc	addi $5,$5,-4	21: addi $a1, $a1, -4 # set lai dia chi cho phan tu cuoi cua mang chua sap xep
□	0x0040003c	0x08100009	j 0x00400024	22: j sort # quay lai chuong trinh con sort
□	0x00400040	0x08100007	j 0x0040001c	23: done : j after_sort # quay lai chuong trinh chinh
□	0x00400044	0x20820000	addi $2,$4,0	25: max: addi $v0, $a0, 0 # con tro chua gia tri max cua mang chua sap xep
□	0x00400048	0x8c430000	lw $3,0($2)	26: lw $v1, 0($v0) # gia tri max cua mang chua sap xep (ban dau bang phan tu dau tien)
□	0x0040004c	0x20880000	addi $8,$4,0	27: addi $t0, $a0, 0 # con tro chay cua mang chua sap xep
□	0x00400050	0x11050008	beq $8,$5,8	28: loop: beq $t0, $a1, ret # neu con tro chay = con tro cuoi cua mang chua sap xep -> ket thuc
□	0x00400054	0x21080004	addi $8,$8,4	29: addi $t0, $t0, 4 # nhay den phan tu ke tiep
□	0x00400058	0x8d090000	lw $9,0($8)	30: lw $t1, 0($t0) # lay gia tri phan tu ke tiep vao t1
□	0x0040005c	0x0123502a	slt $10,$9,$3	31: slt $t2, $t1, $v1
□	0x00400060	0x1540fffb	bne $10,$0,-5	32: bne $t2, $zero, loop # neu phan tu ke tiep < max -> quay lai vong lap
□	0x00400064	0x21020000	addi $2,$8,0	33: addi $v0, $t0, 0 #neu phan tu ke tiep > max -> dia chi max = dia chi phan tu ke tiep
□	0x00400068	0x21230000	addi $3,$9,0	34: addi $v1, $t1, 0 # max = gia tri phan tu ke tiep

Label	Address ▲
mips2_1.asm	
main	0x00400000
after_sort	0x0040001c
end_main	0x00400024
sort	0x00400024
after_max	0x0040002c
done	0x00400040
max	0x00400044
loop	0x00400050
ret	0x00400074
A	0x10010000
Aend	0x10010034

☑ Data ☑ Text

Data Segment

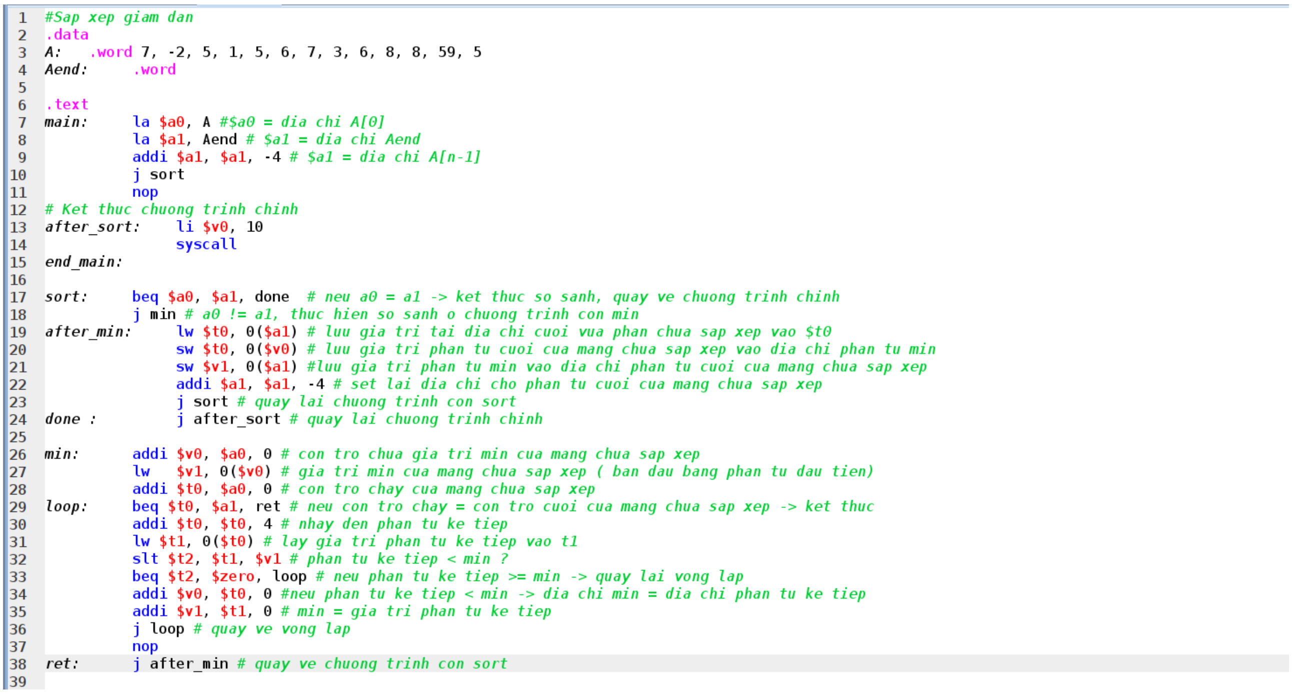
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	-2	1	3	5	5	5	6	6
0x10010020	7	7	8	8	59	0	0	0
0x10010040	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0
0x100100e0	0	0	0	0	0	0	0	0
0x10010100	0	0	0	0	0	0	0	0
0x10010120	0	0	0	0	0	0	0	0
0x10010140	0	0	0	0	0	0	0	0
0x10010160	0	0	0	0	0	0	0	0
0x10010180	0	0	0	0	0	0	0	0
0x100101a0	0	0	0	0	0	0	0	0
0x100101c0	0	0	0	0	0	0	0	0
0x100101e0	0	0	0	0	0	0	0	0

0x10010000 (.data) ☑ Hexadecimal Addresses □ Hexadecimal Values □ ASCII

$zero	0	0
$at	1	268500992
$v0	2	10
$v1	3	1
$a0	4	268500992
$a1	5	268500992
$a2	6	0
$a3	7	0
$t0	8	-2
$t1	9	-2
$t2	10	1
$t3	11	0
$t4	12	0
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	0
$s2	18	0
$s3	19	0
$s4	20	0
$s5	21	0
$s6	22	0
$s7	23	0
$t8	24	0
$t9	25	0
$k0	26	0
$k1	27	0
$gp	28	268468224
$sp	29	2147479548
$fp	30	0
$ra	31	0
pc		4194340
hi		0
lo		0

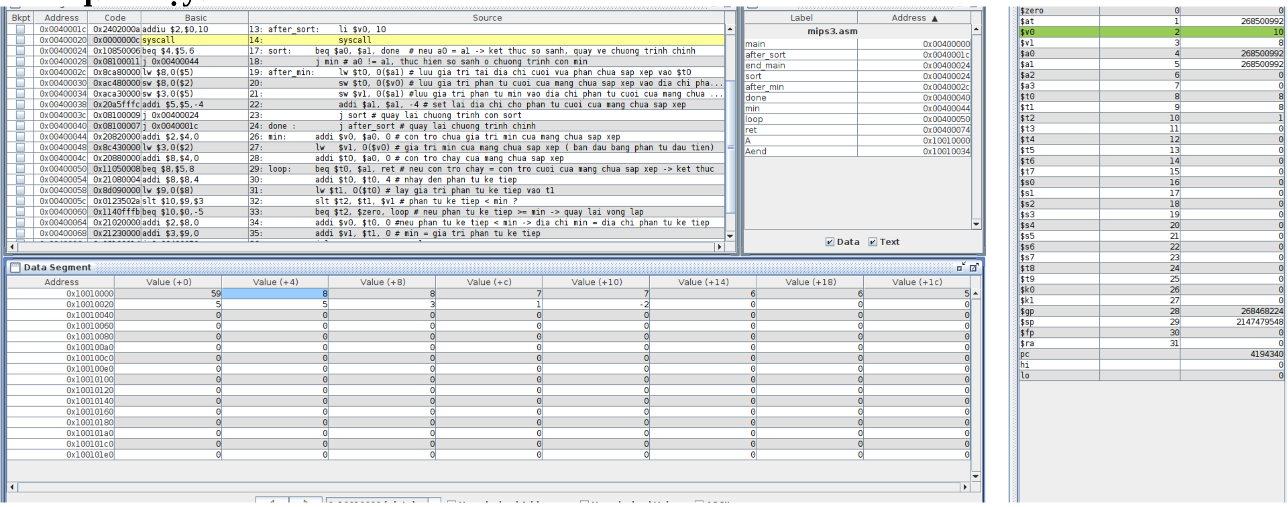
b, Sắp xếp giảm dần:

*Code:



```
#Sap xep giam dan
.data
A:	.word 7, -2, 5, 1, 5, 6, 7, 3, 6, 8, 8, 59, 5
Aend:	.word

.text
main:	la $a0, A #$a0 = dia chi A[0]
	la $a1, Aend # $a1 = dia chi Aend
	addi $a1, $a1, -4 # $a1 = dia chi A[n-1]
	j sort
	nop
# Ket thuc chuong trinh chinh
after_sort:	li $v0, 10
		syscall
end_main:

sort:	beq $a0, $a1, done  # neu a0 = a1 -> ket thuc so sanh, quay ve chuong trinh chinh
	j min # a0 != a1, thuc hien so sanh o chuong trinh con min
after_min:	lw $t0, 0($a1) # luu gia tri tai dia chi cuoi vua phan chua sap xep vao $t0
		sw $t0, 0($v0) # luu gia tri phan tu cuoi cua mang chua sap xep vao dia chi phan tu min
		sw $v1, 0($a1) #luu gia tri phan tu min vao dia chi phan tu cuoi cua mang chua sap xep
		addi $a1, $a1, -4 # set lai dia chi cho phan tu cuoi cua mang chua sap xep
		j sort # quay lai chuong trinh con sort
done :		j after_sort # quay lai chuong trinh chinh

min:	addi $v0, $a0, 0 # con tro chua gia tri min cua mang chua sap xep
	lw   $v1, 0($v0) # gia tri min cua mang chua sap xep ( ban dau bang phan tu dau tien)
	addi $t0, $a0, 0 # con tro chay cua mang chua sap xep
loop:	beq $t0, $a1, ret # neu con tro chay = con tro cuoi cua mang chua sap xep -> ket thuc
	addi $t0, $t0, 4 # nhay den phan tu ke tiep
	lw $t1, 0($t0) # lay gia tri phan tu ke tiep vao t1
	slt $t2, $t1, $v1 # phan tu ke tiep < min ?
	beq $t2, $zero, loop # neu phan tu ke tiep >= min -> quay lai vong lap
	addi $v0, $t0, 0 #neu phan tu ke tiep < min -> dia chi min = dia chi phan tu ke tiep
	addi $v1, $t1, 0 # min = gia tri phan tu ke tiep
	j loop # quay ve vong lap
	nop
ret:	j after_min # quay ve chuong trinh con sort
```

*Kết quả chạy:



Bkpt	Address	Code	Basic	Source
□	0x0040001c	0x2402000a	addiu $2,$0,10	13: after_sort: li $v0, 10
□	0x00400020	0x0000000c	syscall	14: syscall
□	0x00400024	0x10850006	beq $4,$5,6	17: sort: beq $a0, $a1, done # neu a0 = a1 -> ket thuc so sanh, quay ve chuong trinh chinh
□	0x00400028	0x08100011	j 0x00400044	18: j min # a0 != a1, thuc hien so sanh o chuong trinh con min
□	0x0040002c	0x8ca80000	lw $8,0($5)	19: after_min: lw $t0, 0($a1) # luu gia tri tai dia chi cuoi vua phan chua sap xep vao $t0
□	0x00400030	0xac480000	sw $8,0($2)	20: sw $t0, 0($v0) # luu gia tri phan tu cuoi cua mang chua sap xep vao dia chi pha...
□	0x00400034	0xaca30000	sw $3,0($5)	21: sw $v1, 0($a1) #luu gia tri phan tu min vao dia chi phan tu cuoi cua mang chua ...
□	0x00400038	0x20a5fffc	addi $5,$5,-4	22: addi $a1, $a1, -4 # set lai dia chi cho phan tu cuoi cua mang chua sap xep
□	0x0040003c	0x08100009	j 0x00400024	23: j sort # quay lai chuong trinh con sort
□	0x00400040	0x08100007	j 0x0040001c	24: done : j after_sort # quay lai chuong trinh chinh
□	0x00400044	0x20820000	addi $2,$4,0	26: min: addi $v0, $a0, 0 # con tro chua gia tri min cua mang chua sap xep
□	0x00400048	0x8c430000	lw $3,0($2)	27: lw $v1, 0($v0) # gia tri min cua mang chua sap xep (ban dau bang phan tu dau tien)
□	0x0040004c	0x20880000	addi $8,$4,0	28: addi $t0, $a0, 0 # con tro chay cua mang chua sap xep
□	0x00400050	0x11050008	beq $8,$5,8	29: loop: beq $t0, $a1, ret # neu con tro chay = con tro cuoi cua mang chua sap xep -> ket thuc
□	0x00400054	0x21080004	addi $8,$8,4	30: addi $t0, $t0, 4 # nhay den phan tu ke tiep
□	0x00400058	0x8d090000	lw $9,0($8)	31: lw $t1, 0($t0) # lay gia tri phan tu ke tiep vao t1
□	0x0040005c	0x0123502a	slt $10,$9,$3	32: slt $t2, $t1, $v1 # phan tu ke tiep < min ?
□	0x00400060	0x1140fffb	beq $10,$0,-5	33: beq $t2, $zero, loop # neu phan tu ke tiep >= min -> quay lai vong lap
□	0x00400064	0x21020000	addi $2,$8,0	34: addi $v0, $t0, 0 #neu phan tu ke tiep < min -> dia chi min = dia chi phan tu ke tiep
□	0x00400068	0x21230000	addi $3,$9,0	35: addi $v1, $t1, 0 # min = gia tri phan tu ke tiep

Label	Address ▲
mips3.asm	
main	0x00400000
after_sort	0x0040001c
end_main	0x00400024
sort	0x00400024
after_min	0x0040002c
done	0x00400040
min	0x00400044
loop	0x00400050
ret	0x00400074
A	0x10010000
Aend	0x10010034

☑ Data ☑ Text

Data Segment

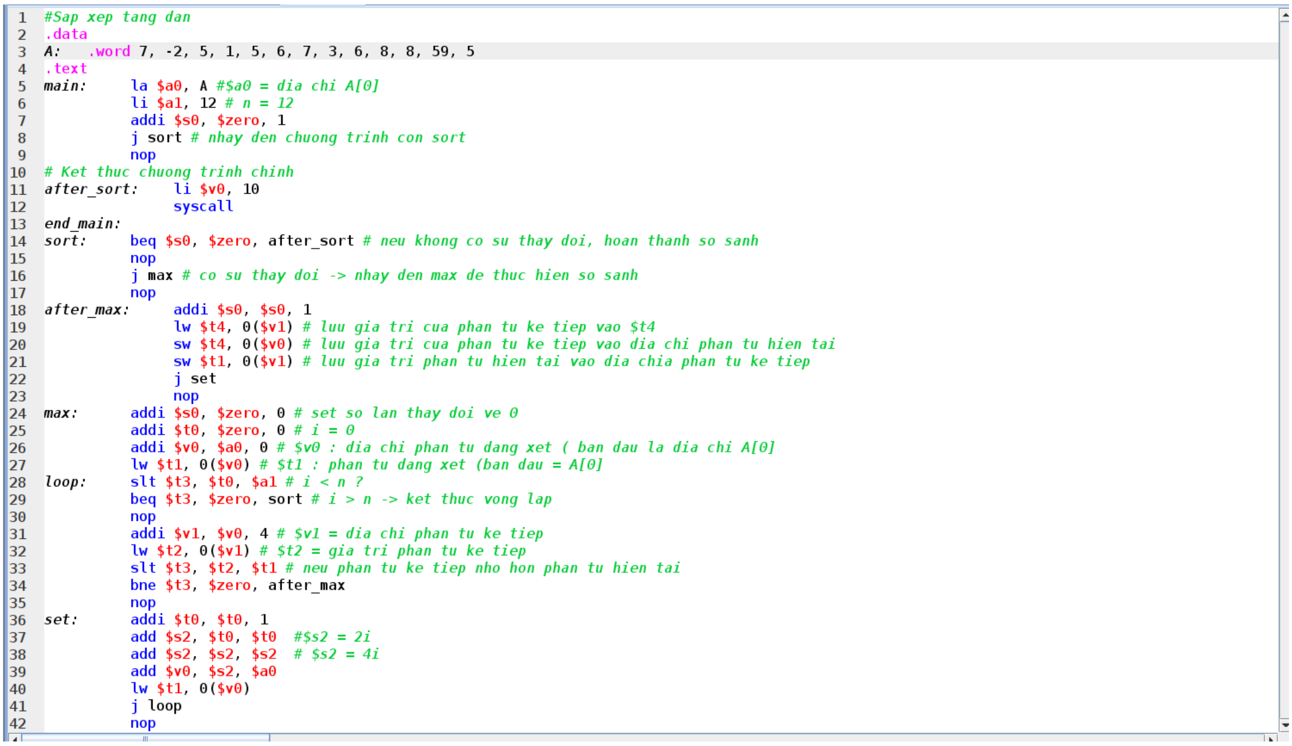
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	59	8	8	7	7	6	6	5
0x10010020	5	5	3	1	-2	0	0	0
0x10010040	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0
0x100100e0	0	0	0	0	0	0	0	0
0x10010100	0	0	0	0	0	0	0	0
0x10010120	0	0	0	0	0	0	0	0
0x10010140	0	0	0	0	0	0	0	0
0x10010160	0	0	0	0	0	0	0	0
0x10010180	0	0	0	0	0	0	0	0
0x100101a0	0	0	0	0	0	0	0	0
0x100101c0	0	0	0	0	0	0	0	0
0x100101e0	0	0	0	0	0	0	0	0

$zero	0	0
$at	1	268500992
$v0	2	10
$v1	3	8
$a0	4	268500992
$a1	5	268500992
$a2	6	0
$a3	7	0
$t0	8	8
$t1	9	8
$t2	10	1
$t3	11	0
$t4	12	0
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	0
$s2	18	0
$s3	19	0
$s4	20	0
$s5	21	0
$s6	22	0
$s7	23	0
$t8	24	0
$t9	25	0
$k0	26	0
$k1	27	0
$gp	28	268468224
$sp	29	2147479548
$fp	30	0
$ra	31	0
pc		4194340
hi		0
lo		0

Bài 3:

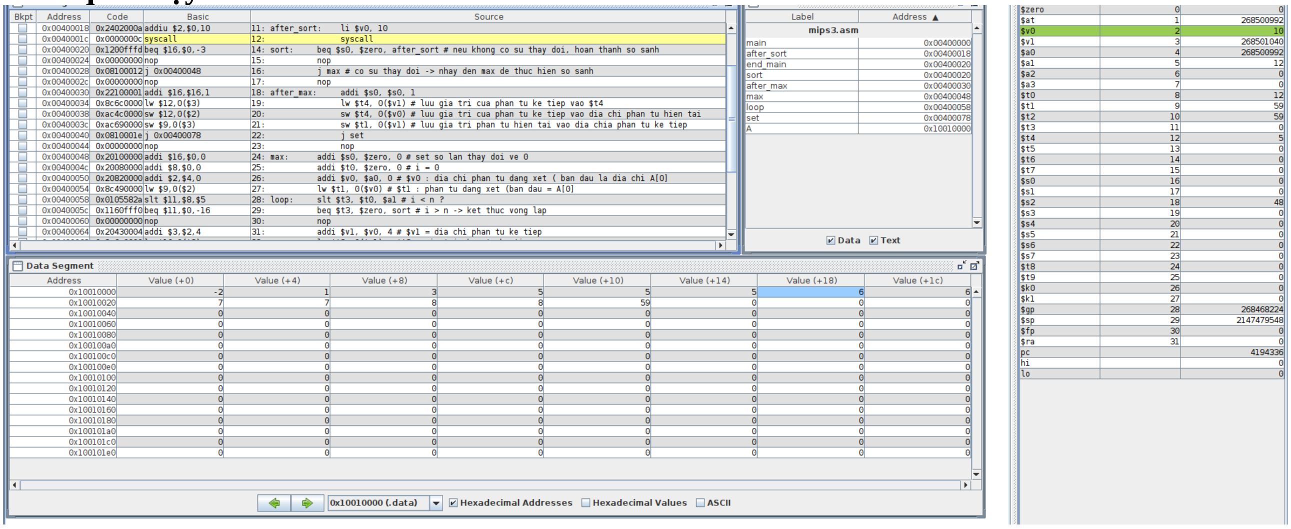
a, Sắp xếp tăng dần:

*Code:



```
#Sap xep tang dan
.data
A:    .word 7, -2, 5, 1, 5, 6, 7, 3, 6, 8, 8, 59, 5
.text
main:      la $a0, A #$a0 = dia chi A[0]
           li $a1, 12 # n = 12
           addi $s0, $zero, 1
           j sort # nhay den chuong trinh con sort
           nop
# Ket thuc chuong trinh chinh
after_sort:     li $v0, 10
                syscall
end_main:
sort:      beq $s0, $zero, after_sort # neu khong co su thay doi, hoan thanh so sanh
           nop
           j max # co su thay doi -> nhay den max de thuc hien so sanh
           nop
after_max:      addi $s0, $s0, 1
                lw $t4, 0($v1) # luu gia tri cua phan tu ke tiep vao $t4
                sw $t4, 0($v0) # luu gia tri cua phan tu ke tiep vao dia chi phan tu hien tai
                sw $t1, 0($v1) # luu gia tri phan tu hien tai vao dia chia phan tu ke tiep
                j set
                nop
max:       addi $s0, $zero, 0 # set so lan thay doi ve 0
           addi $t0, $zero, 0 # i = 0
           addi $v0, $a0, 0 # $v0 : dia chi phan tu dang xet ( ban dau la dia chi A[0]
           lw $t1, 0($v0) # $t1 : phan tu dang xet (ban dau = A[0]
loop:      slt $t3, $t0, $a1 # i < n ?
           beq $t3, $zero, sort # i > n -> ket thuc vong lap
           nop
           addi $v1, $v0, 4 # $v1 = dia chi phan tu ke tiep
           lw $t2, 0($v1) # $t2 = gia tri phan tu ke tiep
           slt $t3, $t2, $t1 # neu phan tu ke tiep nho hon phan tu hien tai
           bne $t3, $zero, after_max
           nop
set:       addi $t0, $t0, 1
           add $s2, $t0, $t0  #$s2 = 2i
           add $s2, $s2, $s2  # $s2 = 4i
           add $v0, $s2, $a0
           lw $t1, 0($v0)
           j loop
           nop
```

*Kết quả chạy:



Bkpt	Address	Code	Basic	Source
□	0x00400018	0x2402000a	addiu $2,$0,10	11: after_sort: li $v0, 10
□	0x0040001c	0x0000000c	syscall	12: syscall
□	0x00400020	0x1200fffd	beq $16,$0,-3	14: sort: beq $s0, $zero, after_sort # neu khong co su thay doi, hoan thanh so sanh
□	0x00400024	0x00000000	nop	15: nop
□	0x00400028	0x08100012	j 0x00400048	16: j max # co su thay doi -> nhay den max de thuc hien so sanh
□	0x0040002c	0x00000000	nop	17: nop
□	0x00400030	0x22100001	addi $16,$16,1	18: after_max: addi $s0, $s0, 1
□	0x00400034	0x8c6c0000	lw $12,0($3)	19: lw $t4, 0($v1) # luu gia tri cua phan tu ke tiep vao $t4
□	0x00400038	0xac4c0000	sw $12,0($2)	20: sw $t4, 0($v0) # luu gia tri cua phan tu ke tiep vao dia chi phan tu hien tai
□	0x0040003c	0xac690000	sw $9,0($3)	21: sw $t1, 0($v1) # luu gia tri phan tu hien tai vao dia chia phan tu ke tiep
□	0x00400040	0x0810001e	j 0x00400078	22: j set
□	0x00400044	0x00000000	nop	23: nop
□	0x00400048	0x20100000	addi $16,$0,0	24: max: addi $s0, $zero, 0 # set so lan thay doi ve 0
□	0x0040004c	0x20080000	addi $8,$0,0	25: addi $t0, $zero, 0 # i = 0
□	0x00400050	0x20820000	addi $2,$4,0	26: addi $v0, $a0, 0 # $v0 : dia chi phan tu dang xet (ban dau la dia chi A[0]
□	0x00400054	0x8c490000	lw $9,0($2)	27: lw $t1, 0($v0) # $t1 : phan tu dang xet (ban dau = A[0]
□	0x00400058	0x0105582a	slt $11,$8,$5	28: loop: slt $t3, $t0, $a1 # i < n ?
□	0x0040005c	0x1160fff0	beq $11,$0,-16	29: beq $t3, $zero, sort # i > n -> ket thuc vong lap
□	0x00400060	0x00000000	nop	30: nop
□	0x00400064	0x20430004	addi $3,$2,4	31: addi $v1, $v0, 4 # $v1 = dia chi phan tu ke tiep

Label	Address ▲
mips3.asm	
main	0x00400000
after_sort	0x00400018
end_main	0x00400020
sort	0x00400020
after_max	0x00400030
max	0x00400048
loop	0x00400058
set	0x00400078
A	0x10010000

☑ Data ☑ Text

$zero	0	0
$at	1	268500992
$v0	2	10
$v1	3	268501040
$a0	4	268500992
$a1	5	12
$a2	6	0
$a3	7	0
$t0	8	12
$t1	9	59
$t2	10	59
$t3	11	0
$t4	12	5
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	0
$s2	18	48
$s3	19	0
$s4	20	0
$s5	21	0
$s6	22	0
$s7	23	0
$t8	24	0
$t9	25	0
$k0	26	0
$k1	27	0
$gp	28	268468224
$sp	29	2147479548
$fp	30	0
$ra	31	0
pc		4194336
hi		0
lo		0

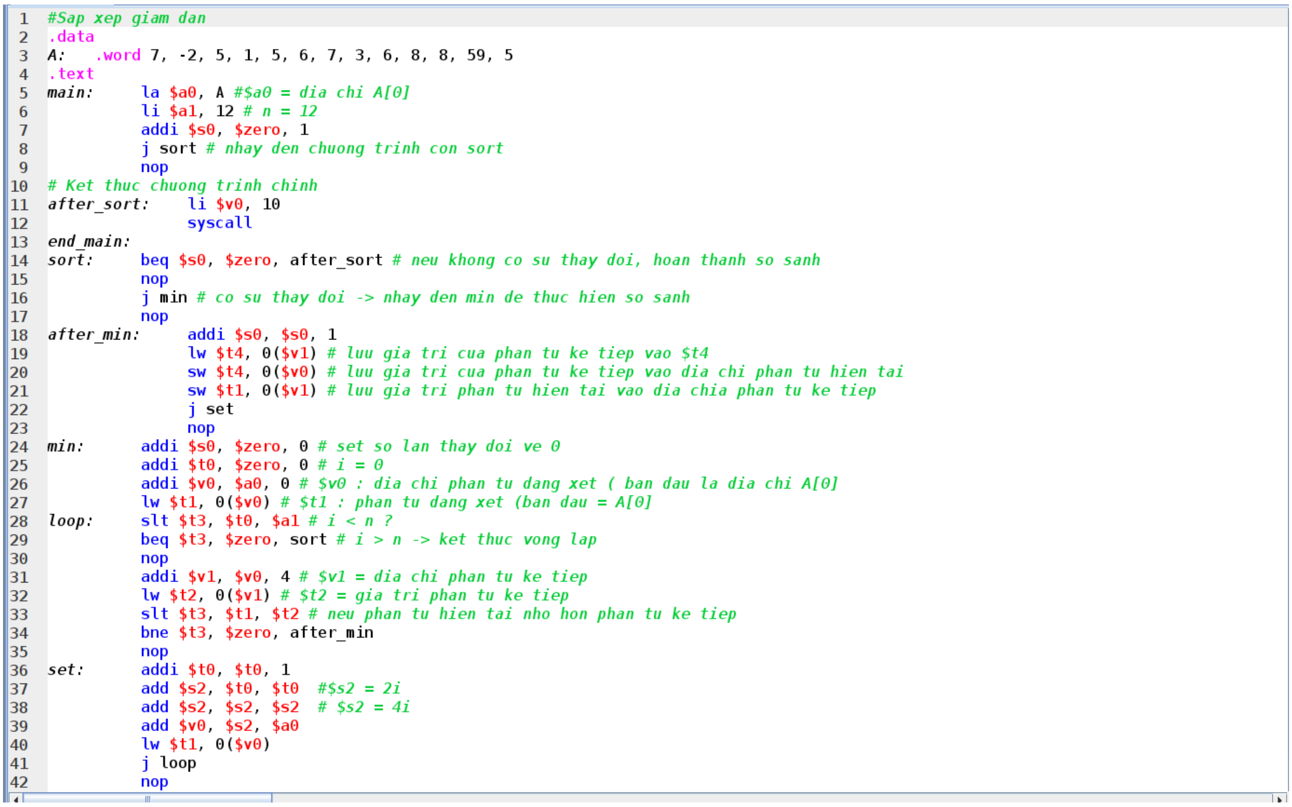
Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	-2	1	3	5	5	5	6	6
0x10010020	7	7	8	8	59	0	0	0
0x10010040	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0
0x100100e0	0	0	0	0	0	0	0	0
0x10010100	0	0	0	0	0	0	0	0
0x10010120	0	0	0	0	0	0	0	0
0x10010140	0	0	0	0	0	0	0	0
0x10010160	0	0	0	0	0	0	0	0
0x10010180	0	0	0	0	0	0	0	0
0x100101a0	0	0	0	0	0	0	0	0
0x100101c0	0	0	0	0	0	0	0	0
0x100101e0	0	0	0	0	0	0	0	0

0x10010000 (.data) ☑ Hexadecimal Addresses □ Hexadecimal Values □ ASCII

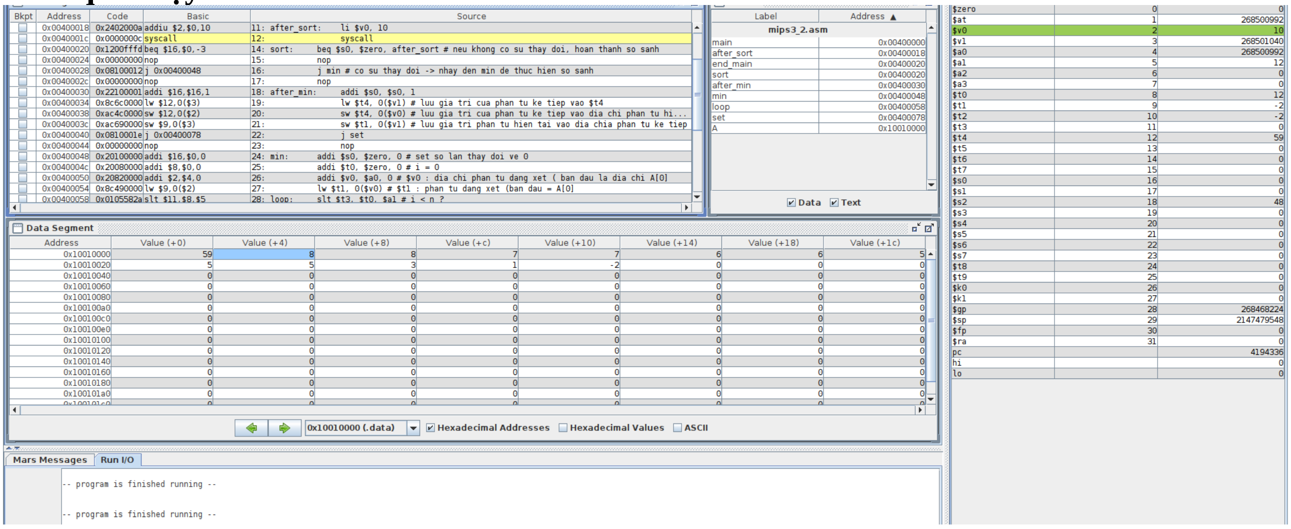
b, Sắp xếp giảm dần:

*Code:



```
#Sap xep giam dan
.data
A:    .word 7, -2, 5, 1, 5, 6, 7, 3, 6, 8, 8, 59, 5
.text
main:       la $a0, A #$a0 = dia chi A[0]
            li $a1, 12 # n = 12
            addi $s0, $zero, 1
            j sort # nhay den chuong trinh con sort
            nop
# Ket thuc chuong trinh chinh
after_sort:     li $v0, 10
                syscall
end_main:
sort:       beq $s0, $zero, after_sort # neu khong co su thay doi, hoan thanh so sanh
            nop
            j min # co su thay doi -> nhay den min de thuc hien so sanh
            nop
after_min:      addi $s0, $s0, 1
                lw $t4, 0($v1) # luu gia tri cua phan tu ke tiep vao $t4
                sw $t4, 0($v0) # luu gia tri cua phan tu ke tiep vao dia chi phan tu hien tai
                sw $t1, 0($v1) # luu gia tri phan tu hien tai vao dia chia phan tu ke tiep
                j set
                nop
min:        addi $s0, $zero, 0 # set so lan thay doi ve 0
            addi $t0, $zero, 0 # i = 0
            addi $v0, $a0, 0 # $v0 : dia chi phan tu dang xet ( ban dau la dia chi A[0]
            lw $t1, 0($v0) # $t1 : phan tu dang xet (ban dau = A[0]
loop:       slt $t3, $t0, $a1 # i < n ?
            beq $t3, $zero, sort # i > n -> ket thuc vong lap
            nop
            addi $v1, $v0, 4 # $v1 = dia chi phan tu ke tiep
            lw $t2, 0($v1) # $t2 = gia tri phan tu ke tiep
            slt $t3, $t1, $t2 # neu phan tu hien tai nho hon phan tu ke tiep
            bne $t3, $zero, after_min
            nop
set:        addi $t0, $t0, 1
            add $s2, $t0, $t0  #$s2 = 2i
            add $s2, $s2, $s2  # $s2 = 4i
            add $v0, $s2, $a0
            lw $t1, 0($v0)
            j loop
            nop
```

*Kết quả chạy:



Bkpt	Address	Code	Basic	Source
☐	0x00400018	0x2402000a	addiu $2,$0,10	11: after_sort: li $v0, 10
☐	0x0040001c	0x0000000c	syscall	12: syscall
☐	0x00400020	0x1200fffd	beq $16,$0,-3	14: sort: beq $s0, $zero, after_sort # neu khong co su thay doi, hoan thanh so sanh
☐	0x00400024	0x00000000	nop	15: nop
☐	0x00400028	0x08100012	j 0x00400048	16: j min # co su thay doi -> nhay den min de thuc hien so sanh
☐	0x0040002c	0x00000000	nop	17: nop
☐	0x00400030	0x22100001	addi $16,$16,1	18: after_min: addi $s0, $s0, 1
☐	0x00400034	0x8c6c0000	lw $12,0($3)	19: lw $t4, 0($v1) # luu gia tri cua phan tu ke tiep vao $t4
☐	0x00400038	0xac4c0000	sw $12,0($2)	20: sw $t4, 0($v0) # luu gia tri cua phan tu ke tiep vao dia chi phan tu hi...
☐	0x0040003c	0xac690000	sw $9,0($3)	21: sw $t1, 0($v1) # luu gia tri phan tu hien tai vao dia chia phan tu ke tiep
☐	0x00400040	0x0810001e	j 0x00400078	22: j set
☐	0x00400044	0x00000000	nop	23: nop
☐	0x00400048	0x20100000	addi $16,$0,0	24: min: addi $s0, $zero, 0 # set so lan thay doi ve 0
☐	0x0040004c	0x20080000	addi $8,$0,0	25: addi $t0, $zero, 0 # i = 0
☐	0x00400050	0x20820000	addi $2,$4,0	26: addi $v0, $a0, 0 # $v0 : dia chi phan tu dang xet (ban dau la dia chi A[0]
☐	0x00400054	0x8c490000	lw $9,0($2)	27: lw $t1, 0($v0) # $t1 : phan tu dang xet (ban dau = A[0]
☐	0x00400058	0x0105582a	slt $11,$8,$5	28: loop: slt $t3, $t0, $a1 # i < n ?

Label	Address ▲
mips3_2.asm	
main	0x00400000
after_sort	0x00400018
end_main	0x00400020
sort	0x00400020
after_min	0x00400030
min	0x00400048
loop	0x00400058
set	0x00400078
A	0x10010000

☑ Data ☑ Text

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	59	8	8	7	7	6	6	5
0x10010020	5	5	3	1	-2	0	0	0
0x10010040	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0
0x100100e0	0	0	0	0	0	0	0	0
0x10010100	0	0	0	0	0	0	0	0
0x10010120	0	0	0	0	0	0	0	0
0x10010140	0	0	0	0	0	0	0	0
0x10010160	0	0	0	0	0	0	0	0
0x10010180	0	0	0	0	0	0	0	0
0x100101a0	0	0	0	0	0	0	0	0
0x100101c0	0	0	0	0	0	0	0	0

0x10010000 (.data) ☑ Hexadecimal Addresses ☐ Hexadecimal Values ☐ ASCII

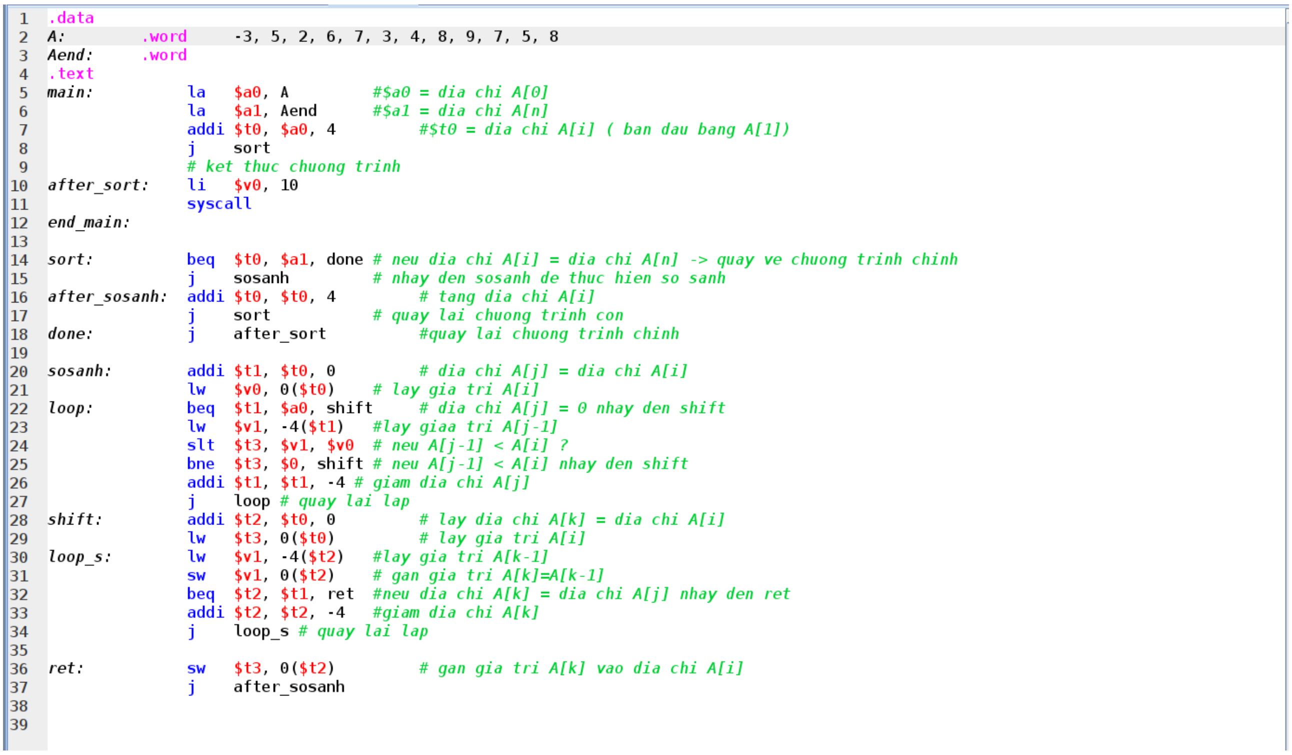
```
-- program is finished running --

-- program is finished running --
```

$zero	0	0
$at	1	268500992
$v0	2	10
$v1	3	268501040
$a0	4	268500992
$a1	5	12
$a2	6	0
$a3	7	0
$t0	8	12
$t1	9	-2
$t2	10	-2
$t3	11	0
$t4	12	59
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	0
$s2	18	48
$s3	19	0
$s4	20	0
$s5	21	0
$s6	22	0
$s7	23	0
$t8	24	0
$t9	25	0
$k0	26	0
$k1	27	0
$gp	28	268468224
$sp	29	2147479548
$fp	30	0
$ra	31	0
pc		4194336
hi		0
lo		0

Bài 4:

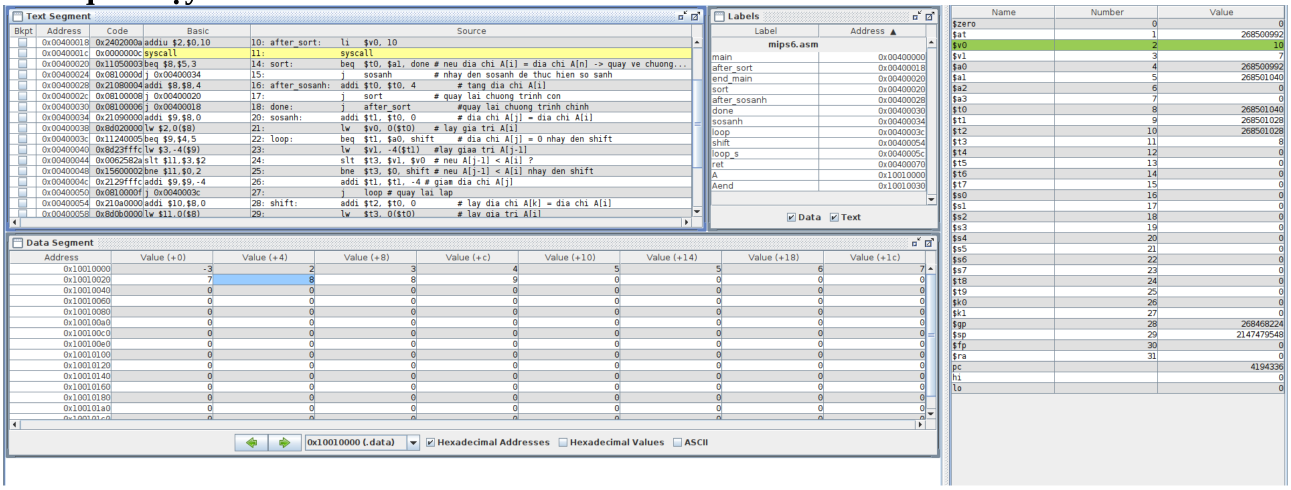
a, Sắp xếp tăng dần:

*Code:



```
.data
A:          .word       -3, 5, 2, 6, 7, 3, 4, 8, 9, 7, 5, 8
Aend:       .word
.text
main:                la   $a0, A         #$a0 = dia chi A[0]
                     la   $a1, Aend      #$a1 = dia chi A[n]
                     addi $t0, $a0, 4         #$t0 = dia chi A[i] ( ban dau bang A[1])
                     j    sort
                     # ket thuc chuong trinh
after_sort:          li   $v0, 10
                     syscall
end_main:

sort:                beq  $t0, $a1, done # neu dia chi A[i] = dia chi A[n] -> quay ve chuong trinh chinh
                     j    sosanh         # nhay den sosanh de thuc hien so sanh
after_sosanh:        addi $t0, $t0, 4         # tang dia chi A[i]
                     j    sort           # quay lai chuong trinh con
done:                j    after_sort          #quay lai chuong trinh chinh

sosanh:              addi $t1, $t0, 0         # dia chi A[j] = dia chi A[i]
                     lw   $v0, 0($t0)    # lay gia tri A[i]
loop:                beq  $t1, $a0, shift     # dia chi A[j] = 0 nhay den shift
                     lw   $v1, -4($t1)   #lay giaa tri A[j-1]
                     slt  $t3, $v1, $v0  # neu A[j-1] < A[i] ?
                     bne  $t3, $0, shift # neu A[j-1] < A[i] nhay den shift
                     addi $t1, $t1, -4 # giam dia chi A[j]
                     j    loop # quay lai lap
shift:               addi $t2, $t0, 0         # lay dia chi A[k] = dia chi A[i]
                     lw   $t3, 0($t0)         # lay gia tri A[i]
loop_s:              lw   $v1, -4($t2)   #lay gia tri A[k-1]
                     sw   $v1, 0($t2)    # gan gia tri A[k]=A[k-1]
                     beq  $t2, $t1, ret  #neu dia chi A[k] = dia chi A[j] nhay den ret
                     addi $t2, $t2, -4   #giam dia chi A[k]
                     j    loop_s # quay lai lap

ret:                 sw   $t3, 0($t2)         # gan gia tri A[k] vao dia chi A[i]
                     j    after_sosanh
```

*Kết quả chạy:



Text Segment

Bkpt	Address	Code	Basic	Source
☐	0x00400018	0x2402000a	addiu $2,$0,10	10: after_sort: li $v0, 10
☐	0x0040001c	0x0000000c	syscall	11: syscall
☐	0x00400020	0x11050003	beq $8,$5,3	14: sort: beq $t0, $a1, done # neu dia chi A[i] = dia chi A[n] -> quay ve chuong...
☐	0x00400024	0x0810000d	j 0x00400034	15: j sosanh # nhay den sosanh de thuc hien so sanh
☐	0x00400028	0x21080004	addi $8,$8,4	16: after_sosanh: addi $t0, $t0, 4 # tang dia chi A[i]
☐	0x0040002c	0x08100008	j 0x00400020	17: j sort # quay lai chuong trinh con
☐	0x00400030	0x08100006	j 0x00400018	18: done: j after_sort #quay lai chuong trinh chinh
☐	0x00400034	0x21090000	addi $9,$8,0	20: sosanh: addi $t1, $t0, 0 # dia chi A[j] = dia chi A[i]
☐	0x00400038	0x8d020000	lw $2,0($8)	21: lw $v0, 0($t0) # lay gia tri A[i]
☐	0x0040003c	0x11240005	beq $9,$4,5	22: loop: beq $t1, $a0, shift # dia chi A[j] = 0 nhay den shift
☐	0x00400040	0x8d23fffc	lw $3,-4($9)	23: lw $v1, -4($t1) #lay giaa tri A[j-1]
☐	0x00400044	0x0062582a	slt $11,$3,$2	24: slt $t3, $v1, $v0 # neu A[j-1] < A[i] ?
☐	0x00400048	0x15600002	bne $11,$0,2	25: bne $t3, $0, shift # neu A[j-1] < A[i] nhay den shift
☐	0x0040004c	0x2129fffc	addi $9,$9,-4	26: addi $t1, $t1, -4 # giam dia chi A[j]
☐	0x00400050	0x0810000f	j 0x0040003c	27: j loop # quay lai lap
☐	0x00400054	0x210a0000	addi $10,$8,0	28: shift: addi $t2, $t0, 0 # lay dia chi A[k] = dia chi A[i]
☐	0x00400058	0x8d0b0000	lw $11,0($8)	29: lw $t3, 0($t0) # lay gia tri A[i]

Labels

Label	Address ▲
mips6.asm	
main	0x00400000
after_sort	0x00400018
end_main	0x00400020
sort	0x00400020
after_sosanh	0x00400028
done	0x00400030
sosanh	0x00400034
loop	0x0040003c
shift	0x00400054
loop_s	0x0040005c
ret	0x00400070
A	0x10010000
Aend	0x10010030

☑ Data ☑ Text

Data Segment

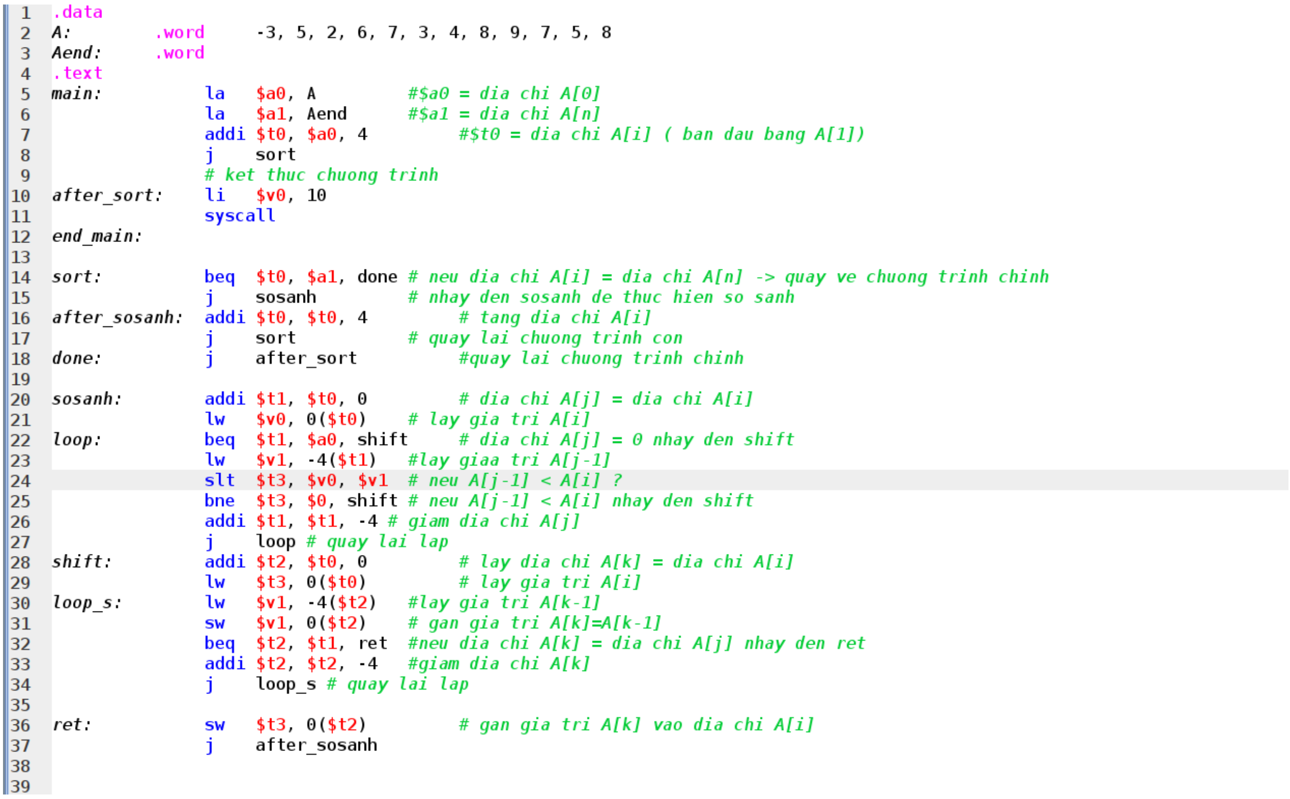
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	-3	2	3	4	5	5	6	7
0x10010020	7	8	8	9	0	0	0	0
0x10010040	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0
0x100100e0	0	0	0	0	0	0	0	0
0x10010100	0	0	0	0	0	0	0	0
0x10010120	0	0	0	0	0	0	0	0
0x10010140	0	0	0	0	0	0	0	0
0x10010160	0	0	0	0	0	0	0	0
0x10010180	0	0	0	0	0	0	0	0
0x100101a0	0	0	0	0	0	0	0	0
0x100101c0	0	0	0	0	0	0	0	0

0x10010000 (.data) ☑ Hexadecimal Addresses ☐ Hexadecimal Values ☐ ASCII

Name	Number	Value
$zero	0	0
$at	1	268500992
$v0	2	10
$v1	3	7
$a0	4	268500992
$a1	5	268501040
$a2	6	0
$a3	7	0
$t0	8	268501040
$t1	9	268501028
$t2	10	268501028
$t3	11	8
$t4	12	0
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	0
$s2	18	0
$s3	19	0
$s4	20	0
$s5	21	0
$s6	22	0
$s7	23	0
$t8	24	0
$t9	25	0
$k0	26	0
$k1	27	0
$gp	28	268468224
$sp	29	2147479548
$fp	30	0
$ra	31	0
pc		4194336
hi		0
lo		0

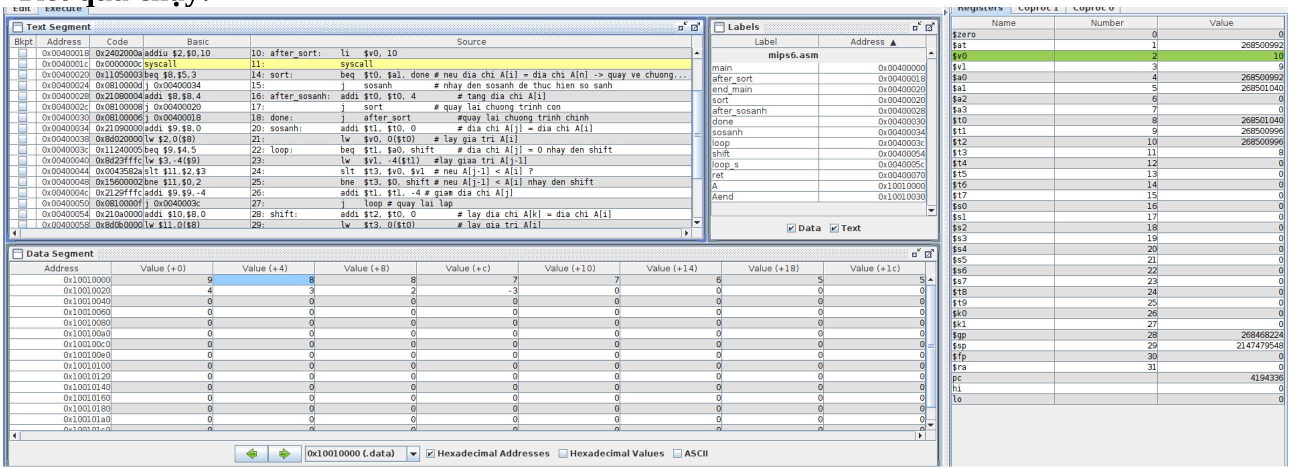
b, Sắp xếp giảm dần:

*Code:



```
.data
A:          .word     -3, 5, 2, 6, 7, 3, 4, 8, 9, 7, 5, 8
Aend:       .word
.text
main:             la    $a0, A         #$a0 = dia chi A[0]
                  la    $a1, Aend      #$a1 = dia chi A[n]
                  addi  $t0, $a0, 4         #$t0 = dia chi A[i] ( ban dau bang A[1])
                  j     sort
                  # ket thuc chuong trinh
after_sort:       li    $v0, 10
                  syscall
end_main:

sort:             beq   $t0, $a1, done # neu dia chi A[i] = dia chi A[n] -> quay ve chuong trinh chinh
                  j     sosanh         # nhay den sosanh de thuc hien so sanh
after_sosanh:     addi  $t0, $t0, 4         # tang dia chi A[i]
                  j     sort           # quay lai chuong trinh con
done:             j     after_sort          #quay lai chuong trinh chinh

sosanh:           addi  $t1, $t0, 0         # dia chi A[j] = dia chi A[i]
                  lw    $v0, 0($t0)    # lay gia tri A[i]
loop:             beq   $t1, $a0, shift     # dia chi A[j] = 0 nhay den shift
                  lw    $v1, -4($t1)   #lay giaa tri A[j-1]
                  slt   $t3, $v0, $v1  # neu A[j-1] < A[i] ?
                  bne   $t3, $0, shift # neu A[j-1] < A[i] nhay den shift
                  addi  $t1, $t1, -4 # giam dia chi A[j]
                  j     loop # quay lai lap
shift:            addi  $t2, $t0, 0         # lay dia chi A[k] = dia chi A[i]
                  lw    $t3, 0($t0)         # lay gia tri A[i]
loop_s:           lw    $v1, -4($t2)   #lay gia tri A[k-1]
                  sw    $v1, 0($t2)    # gan gia tri A[k]=A[k-1]
                  beq   $t2, $t1, ret  #neu dia chi A[k] = dia chi A[j] nhay den ret
                  addi  $t2, $t2, -4   #giam dia chi A[k]
                  j     loop_s # quay lai lap

ret:              sw    $t3, 0($t2)         # gan gia tri A[k] vao dia chi A[i]
                  j     after_sosanh
```

*Kết quả chạy:



Text Segment

Bkpt	Address	Code	Basic	Source
□	0x00400018	0x2402000a	addiu $2,$0,10	10: after_sort: li $v0, 10
□	0x0040001c	0x0000000c	syscall	11: syscall
□	0x00400020	0x11050003	beq $8,$5,3	14: sort: beq $t0, $a1, done # neu dia chi A[i] = dia chi A[n] -> quay ve chuong...
□	0x00400024	0x0810000d	j 0x00400034	15: j sosanh # nhay den sosanh de thuc hien so sanh
□	0x00400028	0x21080004	addi $8,$8,4	16: after_sosanh: addi $t0, $t0, 4 # tang dia chi A[i]
□	0x0040002c	0x08100008	j 0x00400020	17: j sort # quay lai chuong trinh con
□	0x00400030	0x08100006	j 0x00400018	18: done: j after_sort #quay lai chuong trinh chinh
□	0x00400034	0x21090000	addi $9,$8,0	20: sosanh: addi $t1, $t0, 0 # dia chi A[j] = dia chi A[i]
□	0x00400038	0x8d020000	lw $2,0($8)	21: lw $v0, 0($t0) # lay gia tri A[i]
□	0x0040003c	0x11240005	beq $9,$4,5	22: loop: beq $t1, $a0, shift # dia chi A[j] = 0 nhay den shift
□	0x00400040	0x8d23fffc	lw $3,-4($9)	23: lw $v1, -4($t1) #lay giaa tri A[j-1]
□	0x00400044	0x0043582a	slt $11,$2,$3	24: slt $t3, $v0, $v1 # neu A[j-1] < A[i] ?
□	0x00400048	0x15600002	bne $11,$0,2	25: bne $t3, $0, shift # neu A[j-1] < A[i] nhay den shift
□	0x0040004c	0x2129fffc	addi $9,$9,-4	26: addi $t1, $t1, -4 # giam dia chi A[j]
□	0x00400050	0x0810000f	j 0x0040003c	27: j loop # quay lai lap
□	0x00400054	0x210a0000	addi $10,$8,0	28: shift: addi $t2, $t0, 0 # lay dia chi A[k] = dia chi A[i]
□	0x00400058	0x8d0b0000	lw $11,0($8)	29: lw $t3, 0($t0) # lay gia tri A[i]

Labels

Label	Address ▲
mips6.asm	
main	0x00400000
after_sort	0x00400018
end_main	0x00400020
sort	0x00400020
after_sosanh	0x00400028
done	0x00400030
sosanh	0x00400034
loop	0x0040003c
shift	0x00400054
loop_s	0x0040005c
ret	0x00400070
A	0x10010000
Aend	0x10010030

☑ Data ☑ Text

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	9	8	8	7	7	6	5	5
0x10010020	4	3	2	-3	0	0	0	0
0x10010040	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0
0x100100e0	0	0	0	0	0	0	0	0
0x10010100	0	0	0	0	0	0	0	0
0x10010120	0	0	0	0	0	0	0	0
0x10010140	0	0	0	0	0	0	0	0
0x10010160	0	0	0	0	0	0	0	0
0x10010180	0	0	0	0	0	0	0	0
0x100101a0	0	0	0	0	0	0	0	0

0x10010000 (.data) ☑ Hexadecimal Addresses □ Hexadecimal Values □ ASCII

Name	Number	Value
$zero	0	0
$at	1	268500992
$v0	2	10
$v1	3	9
$a0	4	268500992
$a1	5	268501040
$a2	6	0
$a3	7	0
$t0	8	268501040
$t1	9	268500996
$t2	10	268500996
$t3	11	8
$t4	12	0
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	0
$s2	18	0
$s3	19	0
$s4	20	0
$s5	21	0
$s6	22	0
$s7	23	0
$t8	24	0
$t9	25	0
$k0	26	0
$k1	27	0
$gp	28	268468224
$sp	29	2147479548
$fp	30	0
$ra	31	0
pc		4194336
hi		0
lo		0