

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA



BÀI TẬP LỚN MÔN HỌC KIẾN TRÚC MÁY TÍNH

ĐỀ 6:

Cho một chuỗi số nguyên 20 phần tử. Sử dụng hợp ngữ assembly MIPS, viết thu tục sắp xếp chuỗi đó theo thứ tự tăng dần theo giải thuật merge sort. Yêu cầu xuất ra từng bước trong quá trình demo.

LỚP L08--- HK 201

Thành viên: 03 – NGÀY NỘP 09/12/2020

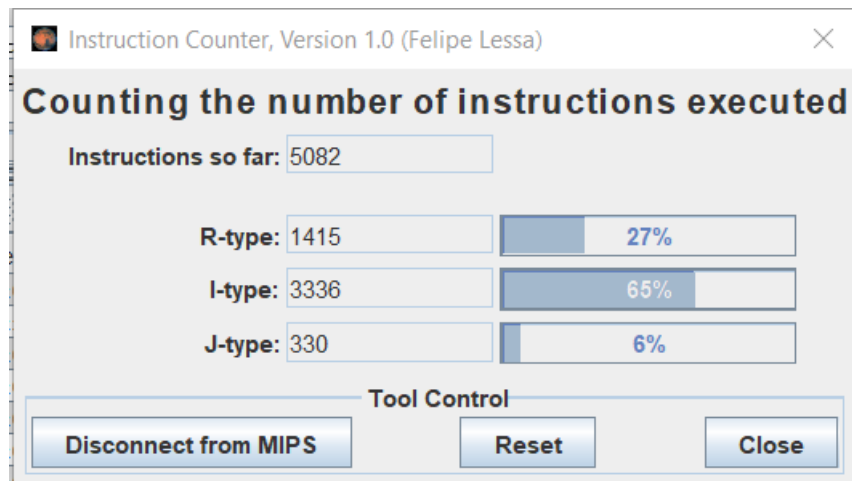
Nguyễn Tấn Đạt – 1911015

Quách Minh Tuấn – 1910663

Võ Văn Tiến Dũng – 1910984

Thành phố Hồ Chí Minh – 2020

I. SUMMARY



Tổng số lệnh là 5082 trong đó:

- 1415 lệnh loại R
- 3336 lệnh loại I
- 330 lệnh loại J

Giả sử chạy chương trình trên máy tính có tần số 2GHz, CPI mỗi lệnh bằng 1

Thời gian thực thi chương trình:
$$\text{Time} = \frac{CPI \times IC}{\text{clockRates}} = \frac{1 \times 5082}{2 \times 10^9} = 2,541 \times 10^{-6} \text{ s}$$

Cấu trúc chương trình:

+ Hàm main(): Ta sẽ khởi tạo các giá trị ban đầu vào các thanh \$a0, \$a1, \$a2

```
9  ##E End declaring
10 .text
11 main:
12 la $a0 iArray #address of first element
13 la $t0 size
14 lw $a1 0($t0) #a1 = 20
15
16 jal mergeSort
17
18 j terminate
19
20 ##### END
```

+ Hàm mergeSort(): Dùng đệ quy để chia nhỏ và dùng **hàm con merge()** sắp xếp mảng

- Chia mảng cần chia ra thành 2 nửa, áp dụng mergeSort lên với mỗi đoạn cho tới khi không chia được nữa thì thôi, tức chỉ có 1 phần tử.

```
30
31 sub $t1, $a1, 1
32 blez $t1, mergeReturn
33 srl $t1, $t1, 1
34 add $t1, $t1, 1
35 sw $t1, 0($sp)
36
37 lw $a1, 0($sp) #a1 = middle + 1
38 #Tai day a1 la middle + 1, a0 la dau mang
39
40
41 jal mergeSort #sort the first half
42
```

```
43 lw $a1, 0($sp)
44 lw $t2, 4($sp)
45 sub $a1, $t2, $a1 # a1 is size of the right half
46
47 lw $a0, 8($sp)
48 lw $t2, 0($sp)
49 sll $t2, $t2, 2
50 add $a0, $a0, $t2 #a0 la dia chi arr[mid + 1]
51
52 jal mergeSort
53
```

- Sau đó merge hai nửa này lại và sắp xếp theo thứ tự tăng dần.

- Cuối cùng, ta in kết quả ra sau khi chạy xong hàm merge.

```

66
67 ### merge
68 jal merge
69
70 ###PRINT ARRAY AFTER MERGE
71 jal printArray
72
73 #Ket thuc goi ham, tra thanh ghi sp ve dieu kien truoc do
74

```

Lặp lại quá trình này cho tới hết khi kết thúc hết mảng

+ Hàm merge(): dùng để gộp 2 mảng con lại và đồng thời sắp xếp chúng

```

#####
merge:
addi $sp $sp -16 #start merge
sw $ra, 12($sp)
sw $a0, 8($sp) #left
sw $a1, 4($sp) #middle
sw $a2, 0($sp) #right

sub $t0 $a1 $a0
addi $t0 $t0 4
srl $t0 $t0 2 #n1

sub $t1 $a2 $a1
srl $t1 $t1 2 #n2

la $t2 iLeft #L[n1]
la $t3 iRight #R[n2]

li $t5, 0 #i

loadL:

bge $t5 $t0 endloop1
sll $t6, $t5, 2
add $t6, $a0, $t6

```

+ Hàm print(): in mảng từ start cho đến end

```

#a0 = left, a1 = mid , a2 = right
printArray:
addi $sp $sp -16
sw $ra, 12($sp)
sw $a0, 8($sp)
sw $a1, 4($sp)
sw $a2, 0($sp)

#la $a0 iArray
sub $t0 $a2 $a0
srl $t0 $t0 2
addi $a1 $t0 1

addi $t0 $0 0 #count
addi $t1 $a0 0 #temporary
loop:
bne $t0 $a1 label

li $v0, 4
la $a0, linebreak
syscall

```

II. TESTCASE (đính kèm kết quả cuối)

Các tiêu chí chọn testcase: Có chứa số 0, số âm, số lớn, số trùng nhau

1/ 69 34 25 28 36 93 95 14 72 44 26 49 38 40 31 53 74 64 82 51

```

34 69
25 34 69
28 36
25 28 34 36 69
93 95
14 93 95
44 72
14 44 72 93 95
14 25 28 34 36 44 69 72 93 95
26 49
26 38 49
31 40
26 31 38 40 49
53 74
53 64 74
51 82
51 53 64 74 82
26 31 38 40 49 51 53 64 74 82
14 25 26 28 31 34 36 38 40 44 49 51 53 64 69 72 74 82 93 95

```

2/ 14 17 9 20 8 16 3 10 6 1 2 4 7 11 19 15 13 12 18 5

```
14 17
9 14 17
8 20
8 9 14 17 20
3 16
3 10 16
1 6
1 3 6 10 16
1 3 6 8 9 10 14 16 17 20
2 4
2 4 7
11 19
2 4 7 11 19
13 15
12 13 15
5 18
5 12 13 15 18
2 4 5 7 11 12 13 15 18 19
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

3/20 24 18 23 27 70 45 74 77 47 56 59 65 93 58 37 48 14 43 68

```
20 24
18 20 24
23 27
18 20 23 24 27
45 70
45 70 74
47 77
45 47 70 74 77
18 20 23 24 27 45 47 70 74 77
56 59
56 59 65
58 93
56 58 59 65 93
37 48
14 37 48
43 68
14 37 43 48 68
14 37 43 48 56 58 59 65 68 93
14 18 20 23 24 27 37 43 45 47 48 56 58 59 65 68 70 74 77 93
```

4/7 -44 -789123456 -49 -60 -12 -36 -85 -61 -64 -46 -71 -76 -96 -86 -9 -41
-4 -81 0

```
-44 7
-789123456 -44 7
-60 -49
-789123456 -60 -49 -44 7
-36 -12
-85 -36 -12
-64 -61
-85 -64 -61 -36 -12
-789123456 -85 -64 -61 -60 -49 -44 -36 -12 7
-71 -46
-76 -71 -46
-96 -86
-96 -86 -76 -71 -46
-41 -9
-41 -9 -4
-81 8
-81 -41 -9 -4 8
-96 -86 -81 -76 -71 -46 -41 -9 -4 8
-789123456 -96 -86 -85 -81 -76 -71 -64 -61 -60 -49 -46 -44 -41 -36 -12 -9 -4 7 8
```

5/3 95 94 35 13 54 29 88 99 42 87 97 19 22 63 10 5 98 50 38

```
3 95
3 94 95
13 35
3 13 35 94 95
29 54
29 54 88
42 99
29 42 54 88 99
3 13 29 35 42 54 88 94 95 99
87 97
19 87 97
22 63
19 22 63 87 97
5 10
5 10 98
38 50
5 10 38 50 98
5 10 19 22 38 50 63 87 97 98
3 5 10 13 19 22 29 35 38 42 50 54 63 87 88 94 95 97 98 99
```

6/40 72 26 80 91 33 55 83 0 6 90 39 84 -53 -52 66 66 66 66 66

```
40 72
26 40 72
80 91
26 40 72 80 91
33 55
33 55 83
0 6
0 6 33 55 83
0 6 26 33 40 55 72 80 83 91
39 90
39 84 90
-53 -52
-53 -52 39 84 90
66 66
66 66 66
66 66
66 66 66 66 66
-53 -52 39 66 66 66 66 66 84 90
-53 -52 0 6 26 33 39 40 55 66 66 66 66 66 72 80 83 84 90 91
```


7/ -4305942 -6578056 4343742 6010410 -8800089 -3150402 5990450 5259747 9662484
3373723 -6965629 -7944070 -2176786 549725 767690 1585445 3896934 1503581 6482672
7481684

```
-6578056 -4305942
-6578056 -4305942 4343742
-8800089 6010410
-8800089 -6578056 -4305942 4343742 6010410
-3150402 5990450
-3150402 5259747 5990450
3373723 9662484
-3150402 3373723 5259747 5990450 9662484
-8800089 -6578056 -4305942 -3150402 3373723 4343742 5259747 5990450 6010410 9662484
-7944070 -6965629
-7944070 -6965629 -2176786
549725 767690
-7944070 -6965629 -2176786 549725 767690
1585445 3896934
1503581 1585445 3896934
6482672 7481684
1503581 1585445 3896934 6482672 7481684
-7944070 -6965629 -2176786 549725 767690 1503581 1585445 3896934 6482672 7481684
-8800089 -7944070 -6965629 -6578056 -4305942 -3150402 -2176786 549725 767690 1503581 1585445 3373723 3896934 4343742 5259747 5990450 6010410 6482672 7481684 9662484
```

8/ -15 -84 60 -45 90 58 69 95 68 26 100 -89 70 30 -76 50 72 -55 -4 66

```
-84 -15
-84 -15 60
-45 90
-84 -45 -15 60 90
58 69
58 69 95
26 68
26 58 68 69 95
-84 -45 -15 26 58 60 68 69 90 95
-89 100
-89 70 100
-76 30
-89 -76 30 70 100
50 72
-55 50 72
-4 66
-55 -4 50 66 72
-89 -76 -55 -4 30 50 66 70 72 100
-89 -84 -76 -55 -45 -15 -4 26 30 50 58 60 66 68 69 70 72 90 95 100
```

9/ -2 -94 -18 96 13 -39 78 -13 10 89 31 -11 -35 -33 16 0 -79 -56 -53 0

```
-94 -2
-94 -18 -2
13 96
-94 -18 -2 13 96
-39 78
-39 -13 78
10 89
-39 -13 10 78 89
-94 -39 -18 -13 -2 10 13 78 89 96
-11 31
-35 -11 31
-33 16
-35 -33 -11 16 31
-79 0
-79 -56 0
-53 0
-79 -56 -53 0 0
-79 -56 -53 -35 -33 -11 0 0 16 31
-94 -79 -56 -53 -39 -35 -33 -18 -13 -11 -2 0 0 10 13 16 31 78 89 96
```

10/ 33 -90 14 44 -34 -58 -85 9 -52 -26 -67 -16 43 75 -81 85 56 62 -57 40

```
-90 33
-90 14 33
-34 44
-90 -34 14 33 44
-85 -58
-85 -58 9
-52 -26
-85 -58 -52 -26 9
-90 -85 -58 -52 -34 -26 9 14 33 44
-67 -16
-67 -16 43
-81 75
-81 -67 -16 43 75
56 85
56 62 85
-57 40
-57 40 56 62 85
-81 -67 -57 -16 40 43 56 62 75 85
-90 -85 -81 -67 -58 -57 -52 -34 -26 -16 9 14 33 40 43 44 56 62 75 85
```