

MỤC LỤC

1.	<i>Thao tác với mảng:</i>	2
2.	<i>Bài tập:</i>	5

BÁO CÁO THỰC HÀNH

Tổ chức & cấu trúc máy tính II – IT012

Bài thực hành 05

Sinh viên: Nguyễn Công Hậu – MSSV: 23520453

Giảng viên hướng dẫn thực hành: Nguyễn Thành Nhân

1. Thao tác với mảng:

a. Code:

```
1  .data
2  array1: .word 5, 6, 7, 8, 1, 2, 3, 9, 10, 4
3  size1:  .word 10
4
5  array2: .byte 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16
6  size2:  .word 16
7
8  array3: .space 8
9  size3:  .word 8
10
11  Nhapm: .ascii "Nhap mang can tim: "
12  Nhapn: .ascii "Nhap vi tri n can tim: "
13  newline: .ascii "\n"
14  space: .ascii " "
15  out_of_bounds: .ascii "Chi so ngoai gioi han\n"
16  invalid_selection: .ascii "Lua chon khong phu hop\n"
17
18  .text
19  main:
20
21  #Lay dia chi nen cua mang, kich thuoc mang, vi tri i = 0
22  la $t0, array1
23  lw $t1, size1
24  li $t2, 0
25
26  #Vong lap for de in mang
27  #so sanh neu dieu kien sai thi dung, mac khac thi thuc hien cau lenh in sau do tang gia tri i va a[i]
28  print_array1_loop:
29      bge $t2, $t1, end_print_array1
30
31      lw $a0, 0($t0)
32
33      li $v0, 1
34      syscall
35
36      li $v0, 11
37      li $a0, ' '
38      syscall
39
40      addi $t0, $t0, 4
41      addi $t2, $t2, 1
42      j print_array1_loop
43
44  end_print_array1:
45
46  #xuong dong
47  li $v0, 11
48  li $a0, '\n'
49  syscall
50
51  #Lay dia chi mang, kich thuoc mang, gia tri i = 0
52  la $s0, array2
53  lw $s1, size2
54  li $s2, 0
55
56  #vong lap for de in ra mang 2
57  #so sanh dieu kien, neu sai thi ket thuc, nguoc lai thi in ra va tang gia tri i, b[i]
58  print_array2_loop:
59      bge $s2, $s1, end_print_array2
60
61      lb $a0, 0($s0)
62
63      li $v0, 1
64      syscall
65
66      li $v0, 11
67      li $a0, ' '
68      syscall
```

```

67     li $a0, ' '
68     syscall
69
70     addi $s0, $s0, 1
71     addi $s2, $s2, 1
72     j print_array2_loop
73
74 end_print_array2:
75
76 #xuống dòng
77 li $v0, 11
78 li $a0, '\n'
79 syscall
80
81 #Lay địa chỉ nên mang2, mang3, kích thước mang2, mang3, giá trị i = 0
82 la $s0, array2
83 lw $s1, size2
84 la $s3, array3
85 lw $s4, size3
86 li $t0, 0
87
88 #vòng lặp để tính giá trị và gán vào mang3
89 compute_array3_loop:
90     # so sánh điều kiện vòng lặp for
91     bge $t0, $s4, end_compute_array3
92
93     #Tính toán giá trị array2[size2 - 1 - i]
94     lb $t3, 0($s0)
95     sub $t4, $s1, $t0
96     subi $t2, $t2, 1
97
98     # Tính array3[i] = array2[i] + array2[size2 - 1 - i]
99     add $t5, $t3, $t4
100
101

```

```

100
101     # Lưu vào mang 3
102     sb $t5, 0($s3)
103
104     # đi chuyển đến phần tử kế tiếp
105     addi $s0, $s0, 1
106     addi $t0, $t0, 1
107     addi $s3, $s3, 1
108     j compute_array3_loop
109
110 end_compute_array3:
111
112 #Tuong tu in mang 1, mang 2
113 #Lay các giá trị địa chỉ nên, kích thước mang, biến i = 0
114 la $s3, array3
115 lw $s2, size3
116 li $s4, 0
117 #Vòng lặp for in mang3
118 print_array3_loop:
119     bge $s4, $s2, end_print_array3
120
121     lb $a0, 0($s3)
122     li $v0, 1
123     syscall
124
125     li $v0, 11
126     li $a0, ' '
127     syscall
128
129     addi $s3, $s3, 1
130     addi $s4, $s4, 1
131     j print_array3_loop
132
133 end_print_array3:

```

```

133 end_print_array3:
134
135 #in xuống dòng
136 li $v0, 11
137 li $a0, '\n'
138 syscall
139
140 #in thông báo nhập m
141 li $v0, 4
142 la $a0, Nhapm
143 syscall
144
145 #Nhập giá trị m
146 li $v0, 5
147 syscall
148 move $s0, $v0
149
150 #in thông báo nhập n
151 li $v0, 4
152 la $a0, Nhapn
153 syscall
154
155 #Nhập giá trị n
156 li $v0, 5
157 syscall
158 move $s1, $v0
159
160 #Kiểm tra điều kiện giống switch case
161 li $t7, 1
162 beq $s0, $t7, case_A
163 li $t7, 2
164 beq $s0, $t7, case_B
165 li $t7, 3
166 beq $s0, $t7, case_C

```

```

166     beq $s0, $t7, case_C
167     j default_case
168
169 case_A:
170
171     #Kiem tra chi so va in mang1
172     li $t8, 10      #Kich thuoc mang1
173     bgeu $s1, $t8, out_of_bounds_case #Neu n >= 10 thi in thong bao khong hop le
174
175     la $t9, array1      # Lay dia chi mang1
176     sll $t0, $s1, 2
177     add $t9, $t9, $t0 #Lay dia chi array1[n]
178     lw $a0, 0($t9) # Lay gia tri array1[n]
179     li $v0, 1
180     syscall
181     j exit
182
183 case_B:
184     #Kiem tra chi so va in mang2
185     li $t8, 16      #Kich thuoc mang2
186     bgeu $s1, $t8, out_of_bounds_case #Neu n >= 16 thi in thong bao khong hop le
187
188     la $t9, array2      #Lay dia chi mang2
189     add $t9, $t9, $s1 #tinh dia chi array2[n]
190     lb $a0, 0($t9) #Lay du lieu array2[n] vao thanh ghi $a0
191     li $v0, 1
192     syscall
193     j exit
194
195 case_C:
196     #kiem tra chi so va in mang3
197     li $t8, 8      #Kich thuoc mang3
198     bgeu $s1, $t8, out_of_bounds_case #Neu n >= 8 thi in ra thong bao khong hop le
199
200     la $t9, array3      #Lay dia chi mang3
201
202     add $t9, $t9, $s1 #Tinh dia chi array3[n]
203     lb $a0, 0($t9) #Lay gia tri array3[n]
204     li $v0, 1
205     syscall
206     j exit
207
208     #in thong bao khong co vi tri phu hop
209 out_of_bounds_case:
210     li $v0, 4
211     la $a0, out_of_bounds
212     syscall
213     j exit
214
215     #in thong bao khong co mang phu hop
216 default_case:
217     li $v0, 4
218     la $a0, invalid_selection
219     syscall
220     j exit
221
222 exit:
223     li $v0, 10
224     syscall
225

```

```

193     j exit
194
195 case_C:
196     #kiem tra chi so va in mang3
197     li $t8, 8      #Kich thuoc mang3
198     bgeu $s1, $t8, out_of_bounds_case #Neu n >= 8 thi in ra thong bao khong hop le
199
200     la $t9, array3      #Lay dia chi mang3
201
202     add $t9, $t9, $s1 #Tinh dia chi array3[n]
203     lb $a0, 0($t9) #Lay gia tri array3[n]
204     li $v0, 1
205     syscall
206     j exit
207
208     #in thong bao khong co vi tri phu hop
209 out_of_bounds_case:
210     li $v0, 4
211     la $a0, out_of_bounds
212     syscall
213     j exit
214
215     #in thong bao khong co mang phu hop
216 default_case:
217     li $v0, 4
218     la $a0, invalid_selection
219     syscall
220     j exit
221
222 exit:
223     li $v0, 10
224     syscall
225

```

b. Kết quả I/O:

Mars MessagesRun I/O

```

5 6 7 8 1 2 3 9 10 4
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
17 17 17 17 17 17 17 17
Nhap mang can tim: 2
Nhap vi tri n can tim: 5
6
-- program is finished running --

```

```

Mars Messages Run I/O
Nhap vi tri n can tim: 5
6
-- program is finished running --

Reset: reset completed.
5 6 7 8 1 2 3 9 10 4
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
17 17 17 17 17 17 17 17
Nhap mang can tim: 1
Nhap vi tri n can tim: 4
1
-- program is finished running --

Mars Messages Run I/O
Nhap vi tri n can tim: 4
1
-- program is finished running --

Reset: reset completed.
5 6 7 8 1 2 3 9 10 4
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
17 17 17 17 17 17 17 17
Nhap mang can tim: 3
Nhap vi tri n can tim: 5
17
-- program is finished running --

```

2. Bài tập:

a. Nhập một mảng các số nguyên n phần tử (nhập vào số phần tử và giá trị của từng phần tử), xuất ra cửa sổ I/O của MARS theo từng yêu cầu sau:

- Xuất ra giá trị lớn nhất và nhỏ nhất của mảng
- Tổng các tất cả các phần tử của mảng
- Người sử dụng nhập vào chỉ số của một phần tử nào đó và giá trị của phần tử đó được in ra cửa sổ

Code:

```

1 .data
2 array: .space 40 #Khai bao mang chua 10 phan tu nguyen, 40 byte
3 size: .word 0
4
5 Nhapn: .asciiz "Nhap kich thuoc mang: "
6 Nhapmang: .asciiz "Nhap mang: "
7 Nhapm: .asciiz "Nhap vi tri can tim gia tri: "
8 newline: .asciiz "\n"
9 space: .asciiz " "
10 error_msg: .asciiz "Error!\n"
11
12 .text
13
14 main:
15     #In thong bao nhap m
16     li $v0, 4
17     la $a0, Nhapn
18     syscall
19
20     #Doc gia tri kich thuoc mang
21     li $v0, 5
22     syscall
23     sw $v0, size
24
25     #Lay dia chi nen, kich thuoc mang va vi tri i
26     la $t0, array
27     lw $t1, size
28     li $t2, 0
29
30     #Thong bao nhap mang

```

```

31     li $v0, 4
32     la $a0, Nhapmang
33     syscall
34
35     #Nhap mang
36     read_array_loop:
37         bge $t2, $t1, end_read_array
38
39         li $v0, 5
40         syscall
41         sw $v0, 0($t0)
42
43         addi $t0, $t0, 4
44         addi $t2, $t2, 1
45         j read_array_loop
46
47     end_read_array:
48
49     #Lay dia chi mang, kich thuoc, i = 0
50     la $t0, array
51     lw $t1, size
52     li $t2, 0
53     #In mang
54     print_array_loop:
55         bge $t2, $t1, end_print_array
56
57         lw $a0, 0($t0)
58         li $v0, 1
59         syscall
60

```

```

61     li $v0, 4
62     la $a0, space
63     syscall
64
65     addi $t0, $t0, 4
66     addi $t2, $t2, 1
67     j print_array_loop
68
69     end_print_array:
70     li $v0, 4
71     la $a0, newline
72     syscall
73
74     #Tim gia tri lon nhat nho nhat
75     la $t0, array
76     lw $t1, 0($t0)    # Gan max = A[0]
77     lw $t2, 0($t0)    # Gan min = A[0]
78
79     lw $t3, size
80     li $t4, 1
81
82     #tim max min
83     find_max_min_loop:
84         bge $t4, $t3, end_find_max_min
85
86         lw $t5, 4($t0) #Gia tri A[i]
87         blt $t1, $t5, update_max #Kiem tra neu lon hon thi gan max, nho hon thi kiem tra va gan min
88         j check_min
89     #Cap nhat gia tri max
90     update_max:

```

```

91     move $t1, $t5
92
93     check_min:
94         bgt $t2, $t5, update_min
95         j next_element
96
97     #Cap nhat gia tri min
98     update_min:
99         move $t2, $t5
100
101     #Nhay den a[i++]
102     next_element:
103         addi $t0, $t0, 4
104         addi $t4, $t4, 1
105         j find_max_min_loop
106
107     end_find_max_min:
108
109     #In max
110     move $a0, $t1
111     li $v0, 1
112     syscall
113     li $v0, 4
114     la $a0, newline
115     syscall
116
117     #In min
118     move $a0, $t2
119     li $v0, 1
120     syscall

```

```

121     li $v0, 4
122     la $a0, newline
123     syscall
124
125     #Tin tong cac phan tu cua mang
126     la $t0, array
127     lw $t1, size
128     li $t2, 0
129     li $t3, 0
130
131 sum_array_loop:
132     bge $t2, $t1, end_sum_array
133
134     lw $t4, 0($t0)
135     add $t3, $t3, $t4
136
137     addi $t0, $t0, 4
138     addi $t2, $t2, 1
139     j sum_array_loop
140
141 end_sum_array:
142
143     #In tong
144     move $a0, $t3
145     li $v0, 1
146     syscall
147     li $v0, 4
148     la $a0, newline
149     syscall
150

```

```

151     #Thong bao nhap vi tri m
152     li $v0, 4
153     la $a0, Nhapm
154     syscall
155
156     #Nhap m
157     li $v0, 5
158     syscall
159     move $s0, $v0
160
161     #In ra gia tri tai vi tri m
162     lw $t1, size
163     bge $s0, $t1, error_case
164     bltz $s0, error_case
165
166     la $t2, array
167     sll $t3, $s0, 2
168     add $t2, $t2, $t3
169     lw $a0, 0($t2)
170     li $v0, 1
171     syscall
172     j exit
173
174 error_case:
175     li $v0, 4
176     la $a0, error_msg
177     syscall
178     j exit
179
180 exit:
181     li $v0, 10
182     syscall
183

```

I/O:

```

Nhap kích thước mảng: 4
Nhap mảng: 0
5
4
1
0 5 4 1
5
0
10
Nhap vị trí cần tìm giá trị: 3
1
--- program is finished running ---

```

b. Nhập một mảng các số nguyên n phần tử (nhập vào số phần tử và giá trị của từng phần tử). Mảng này gọi là A

Chuyển dòng lệnh C dưới đây sang mã assembly của MIPS. Với các biến nguyên i, j được gán lần lượt vào thanh ghi $ss0, ss1$; và địa chỉ nền của mảng số nguyên A được lưu trong thanh ghi $ss3$

if (i<j)

A[i]= i

else

A[i] = j;

Code:

```
1  .data
2  array: .space 40 #Khai bao mang chua 10 phan tu nguyen, 40 byte
3  size: .word 0
4
5  Nhapn: .ascii "Nhap kich thuoc mang: "
6  Nhapmang: .ascii "Nhap mang: "
7  Nhapi: .ascii "Nhap i: "
8  Nhapij: .ascii "Nhap j: "
9  newline: .ascii "\n"
10 space: .ascii " "
11 error_msg: .ascii "Error!\n"
12
13 .text
14
15 main:
16
17     #in thong bao nhap m
18     li $v0, 4
19     la $a0, Nhapi
20     syscall
21
22     #Doc gia tri kich thuoc mang
23     li $v0, 5
24     syscall
25     sw $v0, size
26
27     #Lay dia chi nen, kich thuoc mang va vi tri i
28     la $t0, array
29     lw $t1, size
30     li $t2, 0
31
32     #Thong bao nhap mang
33     li $v0, 4
34     la $a0, Nhapmang
35     syscall
36
37     #Nhap mang
38     read_array_loop:
39         bge $t2, $t1, end_read_array
40
```



```

41     li $v0, 5
42     syscall
43     sw $v0, 0($t0)
44
45     addi $t0, $t0, 4
46     addi $t2, $t2, 1
47     j read_array_loop
48
49 end_read_array:
50
51 #Lay dia chi, kich thuoc, i = 0
52     la $s0, array
53     lw $t1, size
54     li $t2, 0
55
56 #In mang
57 print_array_loop:
58     bge $t2, $t1, end_print_array
59
60     lw $a0, 0($s0)
61     li $v0, 1
62     syscall
63
64     li $v0, 4
65     la $a0, space
66     syscall
67
68     addi $s0, $s0, 4
69     addi $t2, $t2, 1
70     j print_array_loop
71
72 end_print_array:
73     li $v0, 4
74     la $a0, newline
75     syscall
76
77 #Thong bao nhap i
78     li $v0, 4
79     la $a0, Nhapi
80     syscall
81
82 #Nhap gia tri i
83     li $v0, 5
84     syscall
85     move $s0, $v0
86
87 # Thong bao nhap j
88     li $v0, 4
89     la $a0, Nhapij
90     syscall
91
92 #Nhap gia tri j
93     li $v0, 5
94     syscall
95     move $s1, $v0
96
97 #Chuyen cau lenh C
98     blt $s0, $s1, if_true #Kiem tra dieu kien i < j
99
100 if_false:
101     # else A[i] = j;
102     sll $t0, $s0, 2          # t0 = i * 4
103     la $t1, array           # t1 = &A
104     add $t1, $t1, $t0        # t1 = &A[i]
105     sw $s1, 0($t1)          # A[i] = j
106     j end_if
107
108 if_true:
109     # A[i] = i;
110     sll $t0, $s0, 2          # t0 = i * 4
111     la $t1, array           # t1 = &A
112     add $t1, $t1, $t0        # t1 = &A[i]
113     sw $s0, 0($t1)          # A[i] = i
114
115 end_if:
116
117 #Tinh A[i]
118     sll $t0, $s0, 2          # t0 = i * 4
119     la $t1, array           # t1 = &A
120     add $t1, $t1, $t0        # t1 = &A[i]

```

```

120     add $t1, $t1, $t0    # t1 = &A[i]
121     lw $a0, 0($t1)      # $a0 = A[i]
122     li $v0, 1
123     syscall
124
125     #Xuống dòng
126     li $v0, 4
127     la $a0, newline
128     syscall
129
130     #Lay dia chi mang, kich thuoc, i = 0
131     la $s0, array
132     lw $t1, size
133     li $t2, 0
134
135     #vong lap in mang
136     print_array_loop1:
137         bge $t2, $t1, end_print_array1
138
139         lw $a0, 0($s0)
140         li $v0, 1
141         syscall
142
143         li $v0, 4
144         la $a0, space
145         syscall
146
147         addi $s0, $s0, 4
148         addi $t2, $t2, 1
149         j print_array_loop1
150
151     end_print_array1:
152         li $v0, 4
153         la $a0, newline
154         syscall
155
156     exit:
157         li $v0, 10
158         syscall
159

```

I/O:

Mars MessagesRun I/O

Nhap kich thuoc mang: 5

Nhap mang: 2

3

5

1

6

2 3 5 1 6

Nhap i: 3

Nhap j: 4

3

2 3 5 3 6

-- program is finished running --

c. Nhập một mảng số nguyên: sắp xếp tăng, giảm:

Code:

```

1  .data
2  array: .space 40
3  size: .word 0
4
5  Nhapn: .ascii "Nhap vao kich thuoc mang: "
6  Nhapmang: .ascii "Nhap mang: \n"
7  xuat_td: .ascii "Mang tang dan: "
8  xuat_gd: .ascii "Mang giam dan: "
9  newline: .ascii "\n"
10 space: .ascii " "
11 error_msg: .ascii "Error!\n"
12
13 .text
14
15 main:
16
17     #Thong bao nhap kich thuoc mang
18     li $v0, 4
19     la $a0, Nhapn
20     syscall
21
22     #Nhap n
23     li $v0, 5
24     syscall
25     sw $v0, size
26
27     #Thong bao nhap mang
28     li $v0, 4
29     la $a0, Nhapmang
30     syscall
31
32     #Lay dia chi nen, kich thuoc mang, i = 0
33     la $s0, array
34     lw $t0, size
35     li $t2, 0
36     #Nhap mang
37     read_array_loop:
38         bge $t2, $t0, end_read_array
39
40         li $v0, 5

```

```

41     syscall
42     sw $v0, 0($s0)
43
44     addi $s0, $s0, 4
45     addi $t2, $t2, 1
46     j read_array_loop
47
48 end_read_array:
49
50     #Lay dia chi nen, kich thuoc mang, i = 0
51     la $s0, array
52     lw $t1, size
53     li $t2, 0
54
55     # Sap xep mang theo thu tu tang dan
56     add $t1, $s0, $0           # t1 = dia chi cua arr[0]
57     sub $t2, $t0, 1
58     sll $t2, $t2, 2
59     add $t2, $s0, $t2         # t2 = dia chi cua arr[n-1]
60 loop5:
61     bge $t1, $t2, endloop5
62     addi $t3, $t1, 4           # $t3 = dia chi cua arr[i+1] (arr[j])
63     sll $t4, $t0, 2
64     add $t4, $s0, $t4         # $t4 = dia chi cua arr[n]
65 loop5_1:
66     bge $t3, $t4, endloop5_1
67     lw $s1, 0($t1)            # arr[i]
68     lw $s2, 0($t3)            # arr[j]
69     bgt $s2, $s1, continue3
70
71     add $s3, $s1, $0
72     sw $s2, 0($t1)
73     sw $s3, 0($t3)
74 continue3:
75     addi $t3, $t3, 4
76     j loop5_1
77 endloop5_1:
78     addi $t1, $t1, 4
79     j loop5
80 endloop5:

```

```

81          # Xuat mang vua sap xep tang dan
82          li $v0, 4
83          la $a0, xuat_td
84          syscall
85
86          add $t1, $s0, $0          # t1 = dia chi cua arr[0]
87          sll $t2, $t0, 2
88          add $t2, $s0, $t2        # t2 = dia chi cua arr[n]
89      loop6:
90          bge $t1, $t2, endloop6
91          lw $t3, 0($t1)           # $t3 = arr[i]
92
93          li $v0, 1
94          move $a0, $t3
95          syscall
96
97          li $v0, 4
98          la $a0, space
99          syscall
100
101          addi $t1, $t1, 4
102          j loop6
103
104      endloop6:
105
106          #Xuong dong
107          li $v0, 4
108          la $a0, newline
109          syscall
110
111          # Xuat mang theo thu tu giam dan
112          li $v0, 4
113          la $a0, xuat_gd
114          syscall
115
116          sub $t1, $t0, 1
117          sll $t1, $t1, 2
118          add $t1, $s0, $t1        # t1 = dia chi cua arr[n -1]
119          add $t2, $s0, $0        # t2 = dia chi cua arr[0]
120      loop7:
121
122          syscall
123
124          li $v0, 4
125          la $a0, space
126          syscall
127
128          addi $t1, $t1, 4
129          j loop6
130
131      endloop6:
132
133          #Xuong dong
134          li $v0, 4
135          la $a0, newline
136          syscall
137
138          # Xuat mang theo thu tu giam dan
139          li $v0, 4
140          la $a0, xuat_gd
141          syscall
142
143          sub $t1, $t0, 1
144          sll $t1, $t1, 2
145          add $t1, $s0, $t1        # t1 = dia chi cua arr[n -1]
146          add $t2, $s0, $0        # t2 = dia chi cua arr[0]
147      loop7:
148          blt $t1, $t2, EXIT
149          lw $t3, 0($t1)           # $t3 = arr[i]
150
151          li $v0, 1
152          move $a0, $t3
153          syscall
154
155          li $v0, 4
156          la $a0, space
157          syscall
158
159          addi $t1, $t1, -4
160          j loop7
161
162      EXIT:

```

I/O:

Mars Messages		Run I/O
<p>Nhap vao kich thuoc mang: 5</p> <p>Nhap mang:</p> <p>2</p> <p>4</p> <p>6</p> <p>3</p> <p>8</p> <p>Mang tang dan: 2 3 4 6 8</p> <p>Mang giam dan: 8 6 4 3 2</p> <p>-- program is finished running (dropped off bottom) --</p>		

d. Nhập một chuỗi, xuất chuỗi đảo ngược:

Code:

```
1 .data
2 buffer: .space 100
3 newline: .asciiz "\n"
4 Nhapchuoi: .asciiz "Nhap chuoi: \n"
5 .text
6
7 main:
8
9     #Thong bao nhap chuoi
10    li $v0, 4
11    la $a0, Nhapchuoi
12    syscall
13
14    #Nhap vao chuoi
15    li $v0, 8
16    la $a0, buffer
17    li $a1, 100
18    syscall
19
20    #Lay do dai cua chuoi
21    la $s0, buffer
22    li $t0, 0
23
24 Tim_Chieu_Dai:
25    lb $t1, 0($s0)    #Lay dia chi nen cua chuoi
26    beq $t1, $zero, Ket_Thuc_Tim_Chieu_Dai #Neu ky tu roi thi ket thuc
27
28    addi $s0, $s0, 1    #Ky tu tiep theo
29    addi $t0, $t0, 1    #Tang bien dem
30    j Tim_Chieu_Dai
31
32 Ket_Thuc_Tim_Chieu_Dai:
33
34    #In dao nguoc
35    la $s0, buffer    # Lay dia chi nen cua chuoi
36    add $s1, $s0, $t0    # Tro den cuoi chuoi
37    addi $s1, $s1, -1    # Tro ky tu cuoi
38
39 In_Dao_Nguoc:
40    blt $t0, 1, Ket_Thuc_In # i < 0, ket thuc
41
42    #In chuoi
43    lb $a0, 0($s1)
44    li $v0, 11
45    syscall
46
47    addi $s1, $s1, -1    # Giam dia chi con tro xuong 1
48    addi $t0, $t0, -1    # Tang bien dem
49    j In_Dao_Nguoc
50
51 Ket_Thuc_In:
52    #Xuong dong
53    li $v0, 4
54    la $a0, newline
55    syscall
56
57    # Ket thuc
58    li $v0, 10
59    syscall
60
```

I/O:

Mars Messages Run I/O

Nhap chuoi:
IT012_Lab05

50baL_210TI

-- program is finished running --