1,Nhập vào một chuỗi, xuất lại chuỗi đó ra màn hình (echo).

Ví dụ: nhap chuoi: Hello

xuat chuoi: Hello

.data

inputstr: .asciiz "nhap chuoi: "

str: .space 16

outputstr: .asciiz "xuat chuoi:"

.text

.globl main

main:

# in ra mh "nhap chuoi: "

addi $v0,$0,4

la $a0,inputstr

syscall

# nhập chuỗi

addi $v0,$0,8

la $a0,str

addi $a1,$0,16

syscall

# in ra mh "xuat chuoi: "

addi $v0,$0,4

la $a0,outputstr

syscall

# in ra mh chuỗi vừa nhập

addi $v0,$0,4

la $a0,str

syscall

2. Nhập vào một ký tự, xuất ra ký tự liền trước và liền sau.

Ví dụ: nhap ky tu: b

ky tu truoc la: a

ky tu sau la: c

.data

str1: .asciiz "nhap ky tu: "

str2: .asciiz "\nky tu truoc la: "

str3: .asciiz "\nky tu sau la: "

.text

.globl main

main:

#in ra mh: "nhap ki tu: "

li $v0,4

la $a0,str1

syscall

#nhap ky tu

li $v0,12

syscall

#luu ki tu dang truoc vao $t0

sub $t0,$v0,1

#luu ky tu dang sau vao $t1

addi $t1,$v0,1

#in ra mh: "ky tu truoc la: "

la $a0,str2

li $v0,4

syscall

#in ra ky tu truoc

li $v0,11

move $a0,$t0

syscall

#in ra mh: "ky tu sau la: "

la $a0,str2

li $v0,4

syscall

#in ra ki tu sau

li $v0,11

move $a0,$t1

syscall

3. Nhập vào một ký tự hoa, in ra ký tự thường.

Ví dụ: nhap ky tu in hoa: A

ky tu in thuong: a

.data

str1: .asciiz "nhap ky tu in hoa: "

str2: .asciiz "\nky tu thuong la: "

.text

.globl main

main:

#in ra mh: "nhap ki tu in hoa: "

li $v0,4

la $a0,str1

syscall

#nhap ky tu in hoa

li $v0,12

syscall

#luu ki tu in thuong vao $t0

addi $t0,$v0,32

#in ra mh: "ky tu in thuong la: "

la $a0,str2

li $v0,4

syscall

#in ra ky tu in thuong

li $v0,11

move $a0,$t0

syscall

4. Nhập từ bàn phím 2 số nguyên, tính tổng, hiệu, tích, thương của 2 số.

Ví dụ: nhap so thu 1: 7

nhap so thu 2: 4

Tong= 11 Hieu= 3 Tich= 28 Thuong= 1 So du= 3

.data

nhap1: .asciiz "Nhap so thu 1: "

nhap2: .asciiz  "Nhap so thu 2: "

tong: .asciiz "Tong = "

hieu: .asciiz "  Hieu = "

tich: .asciiz "  Tich = "

thuong: .asciiz "   Thuong = "

du: .asciiz "  So du = "

.text

.globl main

main:

# in ra mh "Nhap so thu 1: "

la $a0,nhap1

add $v0,$0,4

syscall

#nhập số thứ nhất

add $v0,$0,5

syscall

add $t0,$0,$v0 #dua so thu nhat vao $t0

# in ra mh "Nhap so thu 2: "

la $a0,nhap2

add $v0,$0,4

syscall

#nhập số thứ 2

add $v0,$0,5

syscall

add $t1,$0,$v0 #dua so thu  2 vao $t1

Tong:

add $t2,$t0,$t1 #tinh tong luu vao $t2

la $a0,tong

add $v0,$0,4

syscall

add $a0,$0,$t2

add $v0,$0,1

syscall

Hieu:

sub $t2,$t0,$t1 #tinh hieu luu vao $t2

la $a0,hieu

add $v0,$0,4

syscall

add $a0,$0,$t2

add $v0,$0,1

syscall

Tich:

la $a0,tich

add $v0,$0,4

syscall

mult $t0,$t1

mflo $t3

add $a0,$0,$t3

add $v0,$0,1

syscall

Thuong:

la $a0,thuong

add $v0,$0,4

syscall

div $t0,$t1

mflo $t3

add $a0,$0,$t3

add $v0,$0,1

syscall

la $a0,du

add $v0,$0,4

syscall

mfhi $t3

add $a0,$0,$t3

add $v0,$0,1

syscall

5. Nhập vào 2 số nguyên, xuất ra số lớn hơn.

Ví dụ: Nhap so thu 1: 6

Nhap so thu 2: 9

so lon nhat la: 9

.data

nhap1: .asciiz "Nhap so thu 1: "

nhap2: .asciiz  "Nhap so thu 2: "

nhap3: .asciiz "so lon nhat la: "

.text

.globl main

main:

# in ra mh "Nhap so thu 1: "

la $a0,nhap1

add $v0,$0,4

syscall

# nhập số thứ nhất

add $v0,$0,5

syscall

add $t0,$0,$v0 #dua so thu nhat vao $t0

# in ra mh "Nhap so thu 2: "

la $a0,nhap2

add $v0,$0,4

syscall

# nhập số thứ 2

add $v0,$0,5

syscall

add $t1,$0,$v0 #dua so thu  2 vao $t1

#in ra mh "so lon nhat la: "

li $v0,4

la $a0,nhap3

syscall

# so sánh 2 số đưa ra số lớn nhất

slt $t2,$t0,$t1

bne $t2,$0,else

move $a0,$t0

li $v0,1

j end

else: move $a0,$t1

li $v0,1

end: syscall

6. Nhập một ký tự từ bàn phím. Nếu ký tự vừa nhập thuộc [0-9], [a-z], [A-Z] thì xuất ra màn hình ký tự đó và loại của ký tự đó (số, chữ thường, chữ hoa).

Ví dụ: nhap ky tu: 5

5 la so

Nhap ky tu : f

f la chu thuong

nhap ky tu : D

D là chu hoa

.data

str0: .asciiz "nhap ky tu: "

str1: .asciiz " la so"

str2: .asciiz " la chu thuong"

str3: .asciiz " la chu hoa"

str4: .asciiz "\nky tu khong hop le nhap lai\n"

.text

.globl main

main:

L: # in ra mh "nhap ky tu: "

li $v0,4

la $a0,str0

syscall

# nhap ky tu

li $v0,12

syscall

add $t2,$0,$v0  # gán ký tự vừa nhập vào $t2

li $t3,48       # $t3 = 48 (số 0 trong mã ascii)

slt $t0,$t3,$t2 #if(48 < $t2) : $t0 = 1 else $t0 = 0

li $t3,57       # $t3 = 57 (số 9 trong mã ascii)

slt $t1,$t2,$t3 # if($t2<57) : $t1 = 1 else $t1 = 1

and $s1,$t0,$t1  # if $t0=1 and $t1=1 : $s1 = 1   else $s1 =0

bne $s1,$0,i    # if $s1 != 0 ($s1 = 1)  goto i

li $t3,97       # $t3 = 97 (chữ a trong mã ascii)

slt $t0,$t3,$t2 #if(97 < $t2) : $t0 = 1 else $t0 = 0

li $t3,122      # $t3 = 122 (chữ z trong mã ascii)

slt $t1,$t2,$t3 # if($t2<122) : $t1 = 1 else $t1 = 1

and $s1,$t0,$t1 # if $t0=1 and $t1=1 : $s1 = 1   else $s1 =0

bne $s1,$0,ii   # if $s1 != 0 ($s1 = 1)  goto ii

li $t3,65       # $t3 = 65 (chữ A trong mã ascii)

slt $t0,$t3,$t2 #if(65 < $t2) : $t0 = 1 else $t0 = 0

li $t3,90       # $t3 = 90 (chữ Z trong mã ascii)

slt $t1,$t2,$t3 # if($t2<90) : $t1 = 1 else $t1 = 1

and $s1,$t0,$t1 # if $t0=1 and $t1=1 : $s1 = 1   else $s1 =0

bne $s1,$0,iii  # if $s1 != 0 ($s1 = 1)  goto iii

beq $s1,$0,A    #if $s1 = 0 goto A

i: # in ra mh "la so"

li $v0,4

la $a0,str1

j  Exit

ii: # in ra mh "la chu thuong"

li $v0,4

la $a0,str2

j  Exit

iii: #in ra mh "la chu hoa"

li $v0,4

la $a0,str3

j  Exit

A: #in ra mh "ky tu khong hop le nhap lai "

li $v0,4

la $a0,str4

syscall

j  L

Exit: syscall

7. Nhập một mảng các số nguyên n phần tử, xuất mảng đó ra màn hình.

Ví dụ: Nhap So Luong Phan Tu Mang: 3

nhap phan tu thu 0 : 1

nhap phan tu thu 1 : 2

nhap phan tu thu 2 : 3

Ket qua:

1 2 3

.data

str1: .asciiz "Nhap So Luong Phan Tu Mang: "

str2: .asciiz "nhap phan tu thu "

str3: .asciiz " : "

tab: .asciiz "\t"

msg\_output: .asciiz "Ket qua:\n"

arr: .word 0:100

.text

.globl main

main:

# in ra mh "Nhap So Luong Phan Tu Mang: "

la $a0,str1

li $v0,4

syscall

# nhap số lượng phần tử

li $v0,5

syscall

move $t0,$v0 # gán số lượng phần tử vào $t0

#khởi tạoo biến đếm cho vòng lặp $t1 = 0

li $t1,0

la $t2,arr

lap: #in ra mh "nhap phan tu thu "

li $v0,4

la $a0,str2

syscall

#in ra mh chỉ số phần tử đang nhập

li $v0,1

move $a0,$t1

syscall

#in ra mh " : "

li $v0,4

la $a0,str3

syscall

#nhập các phần tử của mảng

li $v0,5

syscall

sw   $v0,($t2)

addi $t1,$t1,1

addi $t2,$t2,4

slt  $t3,$t1,$t0 #kiểm tra điều kiện dừng $t1(biến đếm)<$t0(số phần tử)

bne  $t3,$0,lap  #if($t1<$t0)  goto lap

#in ra mh "Ket qua:\n"

li $v0,4

la $a0,msg\_output

syscall

#khởi tạo biến đếm cho vòng lặp $t1 = 0

li $t1,0

la $t2,arr

in: # in các phần tử ra mh

lw  $t4,($t2)

li $v0,1

move $a0,$t4

syscall

# in khoảng tab giữa các phần tử

li $v0,4

la $a0,tab

syscall

addi $t1,$t1,1

addi $t2,$t2,4

slt $t3,$t1,$t0 #kiểm tra điều kiện dừng $t1(biến đếm)<$t0(số phần tử)

bne $t3,$0,in   #if($t1<$t0)  goto in

8. Nhập vào một số nguyên n, tính tổng từ 1 đến n.

Ví dụ: Nhap n : 4

Tong: 10

.data

str1: .asciiz "nhap n: "

str2: .asciiz "tong: "

.text

.globl main

main:

# in ra mh "nhap n: "

li $v0,4

la $a0,str1

syscall

# nhập số phần tử của mảng (n)

li $v0,5

syscall

move $t0,$v0  # gán số phần tử vào $t0

li   $t1,0    #khởi tạo biến đếm i = 0, lưu vào $t1

li   $t3,0    #khởi tạo tổng s = 0,lưu vào $t3

Loop:

addi $t1,$t1,1    # i=i+1

add  $t3,$t3,$t1  # s=s+i

slt  $t2,$t1,$t0  #điều kiện lặp (i<n)

bne  $t2,$0,Loop

# in ra mh "tong: "

li $v0,4

la $a0,str2

syscall

# in ra mh tổng từ 1 đến n

li   $v0,1

move $a0,$t3

syscall

9. Nhập vào một chuỗi. Tính chiều dài của chuỗi.

Ví dụ: Nhap mot chuoi: HCMUS

Chieu dai cua chuoi: 5

.data

str1: .asciiz "nhap chuoi: "

str: .space 16

str2: .asciiz "do dai chuoi chuoi: "

.text

.globl main

main:

#in ra mh "nhap chuoi: "

addi $v0,$0,4

la $a0,str1

syscall

#nhập vào 1 chuỗi

addi $v0,$0,8

la $a0,str

addi $a1,$0,16

syscall

move  $t2,$a0 #gán chuỗi vào $t2

#in ra mh "do dai chuoi: "

li $v0,4

la $a0,str2

syscall

li $t1,0  #khởi tạo biến đếm i = 0, lưu vào $t1

in:

# vòng lăp đếm độ dài chuỗi

lb  $t4,($t2)

addi $t2,$t2,1

addi $t1,$t1,1

bne $t4,$0,in

# in ra giá trị độ dài chuỗi

li $v0,1

subi $a0,$t1,2 # loại b 2 ký tự cuối chuỗi là \n,\0

syscall

Bài 9:

.data

chuoi: .space 100

stNhap: .asciiz "Nhap chuoi can tinh :"

stResult: .asciiz "\nChieu dai cua chuoi = "

.text

la $a0, stNhap #

li $v0, 4 # $v0 = 4

syscall

la $a0, chuoi #

li $v0, 8 # $v0 = 8

la $a1, 100 #

syscall

move $s0, $a0 # $s0 = $a0 chuỗi

addi $t0, $0, 0 # $t0 = $0 + 0 biến đếm

do:

lb $t1, 0($s0) # gan $t1 cho ki tu dau cua chuoi

beq $t1, $0, ndo # if $t1 == $0 then ndo

addi $s0, $s0, 1 # $s0 = $s0 + 1

addi $t0, $t0, 1 # $t0 = $t0 + 1

j do # jump to do

ndo:

la $a0, stResult #

li $v0, 4 # $v0 = 4

syscall

addi $t0, $t0, -1 # $t0 = $t0 + -1

move $a0, $t0 #

li $v0, 1 # $v0 = 1

syscall

exit:

Bài 10:

.data

array: .word 100

stNhap: .asciiz "Nhap so luong mang:"

stNhapItem: .asciiz "\nNhap i:"

stXuongDong: .asciiz "\n"

chuoiMin: .asciiz "Gia tri nho nhat :"

chuoiMax: .asciiz "Gia tri lon nhat :"

.text

# Xuat ra chuoi yeu cau nhap

la $a0,stNhap #

li $v0,4 # $v0 =4

syscall

# Doc gia tri integer tu nguoi nhap

li $v0, 5 # $v0 = 5

syscall

# Gan lai gia tri vua nhap cho $s0 (do dai chuoi)

add $s0, $v0, $0 # $s0 = $v0 + $0

# Gan lai gia tri vua nhap cho $s1 (do dai chuoi)

move $s1, $s0 # $s1 = $s0

# gan dia chi mang cho $t0

la $t0, array # $t0 = array

do:

# dieu kien dung

beq $s1, 0, ndo # if $s1 == 0 then ndo

# Xuat ra chuoi nhap phan tu mang

la $a0, stNhapItem #

li $v0, 4 # $v0 = 4

syscall

# Doc gia tri integer

li $v0, 5 # $v0 = 5

syscall

# gan gia tri vao mang

sw $v0, 0($t0) #

# giam bien chay

addi $s1, $s1, -1 # $s1 = $s1 + -1

# tang chi so mang

addi $t0, $t0, 4 # $t0 = $t0 + 4

j do # jump to do

ndo:

# gan dia chi mang cho $t0

la $t0, array #

# lay ra gia tri dau tien cua mang de so sanh voi cac gia tri sau do

lw $t1, 0($t0) #

# tang chi so mang

addi $t0, $t0, 4 # $t0 = $t0 + 4

# gan do dai mang cho $s1

move $s1, $s0 # $s1 = $s0

# giam bien chay

addi $s1, $s1, -1 # $s1 = $s1 + -1

min:

# dieu kien dung vong lap

beq $s1, 0, ndo1 # if $s1 == 0 then next

# lay ra gia tri tiep theo cua mang

lw $t2, 0($t0) #

addi $t0, $t0, 4 # $t0 = $t0 + 4

addi $s1, $s1, -1 # $s1 = $s1 + -1

# so sang gia tri tiep theo voi gia tri dau cua mang

slt $t3,$t1,$t2

bgtz $t3,min

# neu gia tri tiep theo nho hon gia tri dau mang thi

# gan gia tri tiep theo cho gia tri dau

move $t1, $t2 # $t1 = $t2

j min # jump to do

#$t1 chua min

ndo1:

# gan dia chi mang cho $t0

la $t0, array #

# lay ra gia tri dau mang

lw $t4, 0($t0) #

addi $t0, $t0, 4 # $t0 = $t0 + 4

move $s1, $s0 # $s1 = $s0

addi $s1, $s1, -1 # $s1 = $s1 + -1

max:

beq $s1, 0, next # if $s1 == 0 then next

lw $t5, 0($t0) #

addi $t0, $t0, 4 # $t0 = $t0 + 4

addi $s1, $s1, -1 # $s1 = $s1 + -1

slt $t6,$t5,$t4

bgtz $t6,max

move $t4, $t5 # $t4 = $t5

j max # jump to max

next:

la $a0, stXuongDong #

li $v0, 4 # $v0 = 4

syscall

#Xuat Min

la $a0, chuoiMin #

li $v0, 4 # $v0 = 4

syscall

move $a0, $t1 # $a0 = $t1

li $v0, 1 # $v0 = 1

syscall

la $a0, stXuongDong #

li $v0, 4 # $v0 = 4

syscall

#Xuat Max

la $a0, chuoiMax #

li $v0, 4 # $v0 = 4

syscall

move $a0, $t4 # $a0 = $t4

li $v0, 1 # $v0 = 1

syscall

exit:

li $v0, 10 # $v0 = 10

syscall

Bài 12:

.data

chuoi: .space 40

label: .asciiz "Nhap mot chuoi:"

label2: .asciiz "Chuoi da nhap:"

.text

main:

la $a0, label # $a0 = label

li $v0, 4 # syscall print str

syscall # execute

la $a0, chuoi #

la $a1, 30 #

li $v0, 8 # $v0 = 8

syscall

move $a1, $a0 # $a1 = $a0

subu $sp,$sp,4

sw $a1, 0($sp) # đẩy địa chỉ vùng nhớ chứa chuỗi vào stack

jal PRINT # jump to PRINT and save position to $ra

exit:

li $v0, 10 # $v0 = 10

syscall

PRINT:

lw $a1, 0($sp) # lấy từ stack ra địa chỉ vùng nhớ chữa chuỗi

addu $sp,$sp,4

la $a0, label2 # $a0 = label2

li $v0, 4 # syscall print str

syscall # execute

move $a0, $a1 # $a0 = $a1

li $v0, 4 # syscall print str

syscall # execute

jr $ra # jump to $ra

Bài 13:

.data

stNhap1: .asciiz "\nNhap so nguyen 1:"

stNhap2: .asciiz "\nNhap so nguyen 2:"

tong: .asciiz "\nTong = "

hieu: .asciiz "\nHieu = "

tich: .asciiz "\nTich = "

thuong: .asciiz "\nThuong = "

du: .asciiz "\nDu = "

.text

main:

subu $sp,$sp,4

sw $t3, ($sp) #

#in ra chuỗi :"Nhap so nguyen 1:"

la $a0, stNhap1 #

jal printf # jump to printf and save position to $ra

# read integer

addi $v0, $0, 5 # $v0 = $0 + 5

syscall

# lưu lại giá trị vừa đọc vào $t0

add $t0, $0, $v0 # $s0 = $v0 + 0

subu $sp,$sp,4

sw $t0, ($sp) #

# in ra màn hình chuỗi : "Nhap so nguyen 2:"

la $a0, stNhap2 #

jal printf # jump to printf and save position to $ra

# read integer

addi $v0, $0, 5 # $v0 = $0 + 5

syscall

# lưu giá trị vừa đọc vào $t1

add $t1, $0, $v0 # $s0 = $v0 + 0

subu $sp,$sp,4

sw $t1, ($sp) #

# tổng

jal SUM # jump to SUM and save position to $ra

la $a0, tong #

jal printResult # jump to printResult and save position to $ra

#hiệu

jal SUB # jump to SUB and save position to $ra

la $a0, hieu #

jal printResult # jump to printResult and save position to $ra

#thương

jal DIVICE # jump to DIVICE and save position to $ra

la $a0, thuong #

jal printResult # jump to printResult and save position to $ra

la $a0, du #

li $v0, 4 # $v0 = 4

syscall

move $a0, $t7 #

li $v0, 1 # $v0 = 1

syscall

#tích

jal MUTL # jump to MUTL and save position to $ra

la $a0, tich #

jal printResult # jump to printResult and save position to $ra

exit:

li $v0, 10 # $v0 = 10

syscall

printResult:

move $t6, $v0 # $t6 = $v0

li $v0, 4 # $v0 = 4

syscall

move $a0, $t6 #

li $v0, 1 # $v0 = 1

syscall

jr $ra # jump to $ra

printf:

li $v0, 4 # $v0 = 4

syscall

jr $ra # jump to $ra

SUM:

lw $t1, ($sp) #

addu $sp,$sp,4

lw $t0, ($sp) #

addu $sp,$sp,4

lw $t3, ($sp) #

addu $sp,$sp,4

add $t3, $t0, $t1 # $t3 = $t0 + $t1

subu $sp,$sp,4

sw $t4, ($sp) #

subu $sp,$sp,4

sw $t1, ($sp) #

subu $sp,$sp,4

sw $t0, ($sp) #

add $v0, $t3, $0 # $v0 = $t3 + $0

jr $ra # jump to $ra

SUB:

lw $t0, ($sp) #

addu $sp,$sp,4

lw $t1, ($sp) #

addu $sp,$sp,4

lw $t4, ($sp) #

addu $sp,$sp,4

sub $t4, $t0, $t1 # $t4 = $t0 - $t1

subu $sp,$sp,4

sw $t5, ($sp) #

subu $sp,$sp,4

sw $t1, ($sp) #

subu $sp,$sp,4

sw $t0, ($sp) #

add $v0, $t4, $0 # $v0 = $t4 + $0

jr $ra # jump to $ra

DIVICE:

lw $t0, ($sp) #

addu $sp,$sp,4

lw $t1, ($sp) #

addu $sp,$sp,4

lw $t5, ($sp) #

addu $sp,$sp,4

div $t0, $t1 # $t0 / $t1

mflo $t5 # $t5 = floor($t0 / $t1)

mfhi $t7 # $t7 = $t0 mod $t1

subu $sp,$sp,4

sw $t1, ($sp) #

subu $sp,$sp,4

sw $t0, ($sp)

add $v0, $t5, $0 # $v0 = $t5 + $0

jr $ra # jump to $ra

MUTL:

lw $t0, ($sp) #

addu $sp,$sp,4

lw $t1, ($sp) #

addu $sp,$sp,4

mult $t0, $t1 # $t0 \* $t1 = Hi and Lo registers

mflo $t8 # copy Lo to $t8

add $v0, $t8, $0 # $v0 = $t8 + $0

jr $ra # jump to $ra

Bài 14:

.data

array: .word 100

stNhap: .asciiz "Nhap so luong mang:"

stNhapItem: .asciiz "\nNhap i:"

stXuongDong: .asciiz "\n"

chuoiMin: .asciiz "Gia tri nho nhat :"

chuoiMax: .asciiz "Gia tri lon nhat :"

.text

# Xuat ra chuoi yeu cau nhap

la $a0,stNhap #

li $v0,4 # $v0 =4

syscall

# Doc gia tri integer tu nguoi nhap

li $v0, 5 # $v0 = 5

syscall

# Gan lai gia tri vua nhap cho $s0 (do dai chuoi)

add $s0, $v0, $0 # $s0 = $v0 + $0

# Gan lai gia tri vua nhap cho $s1 (do dai chuoi)

move $s1, $s0 # $s1 = $s0

# gan dia chi mang cho $t0

la $t0, array # $t0 = array

do:

# dieu kien dung

beq $s1, 0, ndo # if $s1 == 0 then ndo

# Xuat ra chuoi nhap phan tu mang

la $a0, stNhapItem #

li $v0, 4 # $v0 = 4

syscall

# Doc gia tri integer

li $v0, 5 # $v0 = 5

syscall

# gan gia tri vao mang

sw $v0, 0($t0) #

# giam bien chay

addi $s1, $s1, -1 # $s1 = $s1 + -1

# tang chi so mang

addi $t0, $t0, 4 # $t0 = $t0 + 4

j do # jump to do

ndo:

# gan dia chi mang cho $t0

la $t0, array #

# lay ra gia tri dau tien cua mang de so sanh voi cac gia tri sau do

lw $t1, 0($t0) #

# tang chi so mang

addi $t0, $t0, 4 # $t0 = $t0 + 4

# gan do dai mang cho $s1

move $s1, $s0 # $s1 = $s0

# giam bien chay

addi $s1, $s1, -1 # $s1 = $s1 + -1

jal MinMax # jump to MinMax and save position to $ra

next:

la $a0, stXuongDong #

li $v0, 4 # $v0 = 4

syscall

#Xuat Min

la $a0, chuoiMin #

li $v0, 4 # $v0 = 4

syscall

move $a0, $t1 # $a0 = $t1

li $v0, 1 # $v0 = 1

syscall

la $a0, stXuongDong #

li $v0, 4 # $v0 = 4

syscall

#Xuat Max

la $a0, chuoiMax #

li $v0, 4 # $v0 = 4

syscall

move $a0, $t4 # $a0 = $t4

li $v0, 1 # $v0 = 1

syscall

exit:

li $v0, 10 # $v0 = 10

syscall

MinMax:

min:

# dieu kien dung vong lap

beq $s1, 0, ndo1 # if $s1 == 0 then next

# lay ra gia tri tiep theo cua mang

lw $t2, 0($t0) #

addi $t0, $t0, 4 # $t0 = $t0 + 4

addi $s1, $s1, -1 # $s1 = $s1 + -1

# so sang gia tri tiep theo voi gia tri dau cua mang

slt $t3,$t1,$t2

bgtz $t3,min

# neu gia tri tiep theo nho hon gia tri dau mang thi

# gan gia tri tiep theo cho gia tri dau

move $t1, $t2 # $t1 = $t2

j min # jump to do

#$t1 chua min

ndo1:

# gan dia chi mang cho $t0

la $t0, array #

# lay ra gia tri dau mang

lw $t4, 0($t0) #

addi $t0, $t0, 4 # $t0 = $t0 + 4

move $s1, $s0 # $s1 = $s0

addi $s1, $s1, -1 # $s1 = $s1 + -1

max:

beq $s1, 0, next # if $s1 == 0 then next

lw $t5, 0($t0) #

addi $t0, $t0, 4 # $t0 = $t0 + 4

addi $s1, $s1, -1 # $s1 = $s1 + -1

slt $t6,$t5,$t4

bgtz $t6,max

move $t4, $t5 # $t4 = $t5

j max # jump to max