**Báo cáo Thực hành KTMT buổi 10**

**Họ và tên:** Nguyễn Đức Phú

**MSSV:** 20215116

**Assignment 1: Hiển thị 2 số cuối MSSV (16)**

* Code:

.eqv SEVENSEG\_LEFT 0xFFFF0011

.eqv SEVENSEG\_RIGHT 0xFFFF0010

.text

main:

#MSSV:20215116

#2 so cuoi cung: 16

li $a0, 0x06

jal SHOW\_7SEG\_LEFT # show

nop

li $a0, 0x7D

jal SHOW\_7SEG\_RIGHT # show

nop

exit: li $v0, 10

syscall

endmain:

#--------------------------------------------------

# Function SHOW\_7SEG\_LEFT : turn on/off the 7seg

# param[in] $a0 value to shown

# remark $t0 changed

#--------------------------------------------------

SHOW\_7SEG\_LEFT: li $t0, SEVENSEG\_LEFT

sb $a0, 0($t0)

nop

jr $ra

nop

#--------------------------------------------------

# Function SHOW\_7SEG\_RIGHT : turn on/off the 7seg

# param[in] $a0 value to shown

# remark $t0 changed

#--------------------------------------------------SHOW\_7SEG\_RIGHT: li $t0, SEVENSEG\_RIGHT

sb $a0, 0($t0) # assign new value

nop

jr $ra

nop

* Kết quả chạy thử:

A screenshot of a computer

Description automatically generated

**Assignment 2: Nhập vào một số nguyên từ bàn phím, hiển thị 2 chữ số cuối của số nguyên đó**

* Code:

.eqv SEVENSEG\_LEFT 0xFFFF0011

.eqv SEVENSEG\_RIGHT 0xFFFF0010

.data

message: .asciiz "Nhap vao mot so nguyen: "

arr: .word 0x3F,0x6,0x5B,0x4F,0x66,0x6D,0x7D,0x7,0x7F,0x6F

#Ma de hien thi tu 0->9

.text

main:

li $v0, 4

la $a0, message

syscall

li $v0, 5

syscall

move $s0, $v0 #s0 = So nhap vao

li $t2, 10

div $s0, $s0, $t2

mfhi $t1 # lay phan du khi chia 10

sll $t1,$t1,2

la $a1, arr

add $a1,$a1,$t1

lw $a0,0($a1) #lay ra ma hien thi

jal SHOW\_7SEG\_RIGHT

#lam lai tuong tu hien thi so con lai

la $a1, arr

div $s0, $s0, $t2

mfhi $t1

sll $t1,$t1,2

la $a1, arr

add $a1,$a1,$t1

lw $a0,0($a1)

jal SHOW\_7SEG\_LEFT

li $v0, 10 #Exit

syscall

#--------------------------

# Function SHOW\_7SEG\_LEFT : turn on/off the 7seg

# param[in] $a0 value to shown

# remark $t0 changed

#--------------------------

SHOW\_7SEG\_LEFT:

li $t0, SEVENSEG\_LEFT

sb $a0, 0($t0)

jr $ra

#---------------------------

# Function SHOW\_7SEG\_RIGHT : turn on/off the 7seg

# param[in] $a0 value to shown

# remark $t0 changed

#--------------------------

SHOW\_7SEG\_RIGHT:

li $t0, SEVENSEG\_RIGHT

sb $a0, 0($t0)

jr $ra

* Kết quả chạy thử khi nhập vào số 1234 và 9:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Assignment 3:**

* Code:

.eqv SEVENSEG\_LEFT 0xFFFF0011

.eqv SEVENSEG\_RIGHT 0xFFFF0010

.data

message: .asciiz "Nhap vao mot ky tu: "

arr: .word 0x3F,0x6,0x5B,0x4F,0x66,0x6D,0x7D,0x7,0x7F,0x6F

#Ma de hien thi tu 0->9

.text

main:

li $v0, 4

la $a0, message

syscall

li $v0, 12

syscall

move $s0, $v0

li $t2, 10

div $s0, $s0, $t2

mfhi $t1 # lay phan du khi chia 10

sll $t1,$t1,2

la $a1, arr

add $a1,$a1,$t1

lw $a0,0($a1) #lay ra ma hien thi

jal SHOW\_7SEG\_RIGHT

#lam lai tuong tu hien thi so con lai

la $a1, arr

div $s0, $s0, $t2

mfhi $t1

sll $t1,$t1,2

la $a1, arr

add $a1,$a1,$t1

lw $a0,0($a1)

jal SHOW\_7SEG\_LEFT

li $v0, 10 #Exit

syscall

#--------------------------

# Function SHOW\_7SEG\_LEFT : turn on/off the 7seg

# param[in] $a0 value to shown

# remark $t0 changed

#--------------------------

SHOW\_7SEG\_LEFT:

li $t0, SEVENSEG\_LEFT

sb $a0, 0($t0)

jr $ra

#---------------------------

# Function SHOW\_7SEG\_RIGHT : turn on/off the 7seg

# param[in] $a0 value to shown

# remark $t0 changed

#--------------------------

SHOW\_7SEG\_RIGHT:

li $t0, SEVENSEG\_RIGHT

sb $a0, 0($t0)

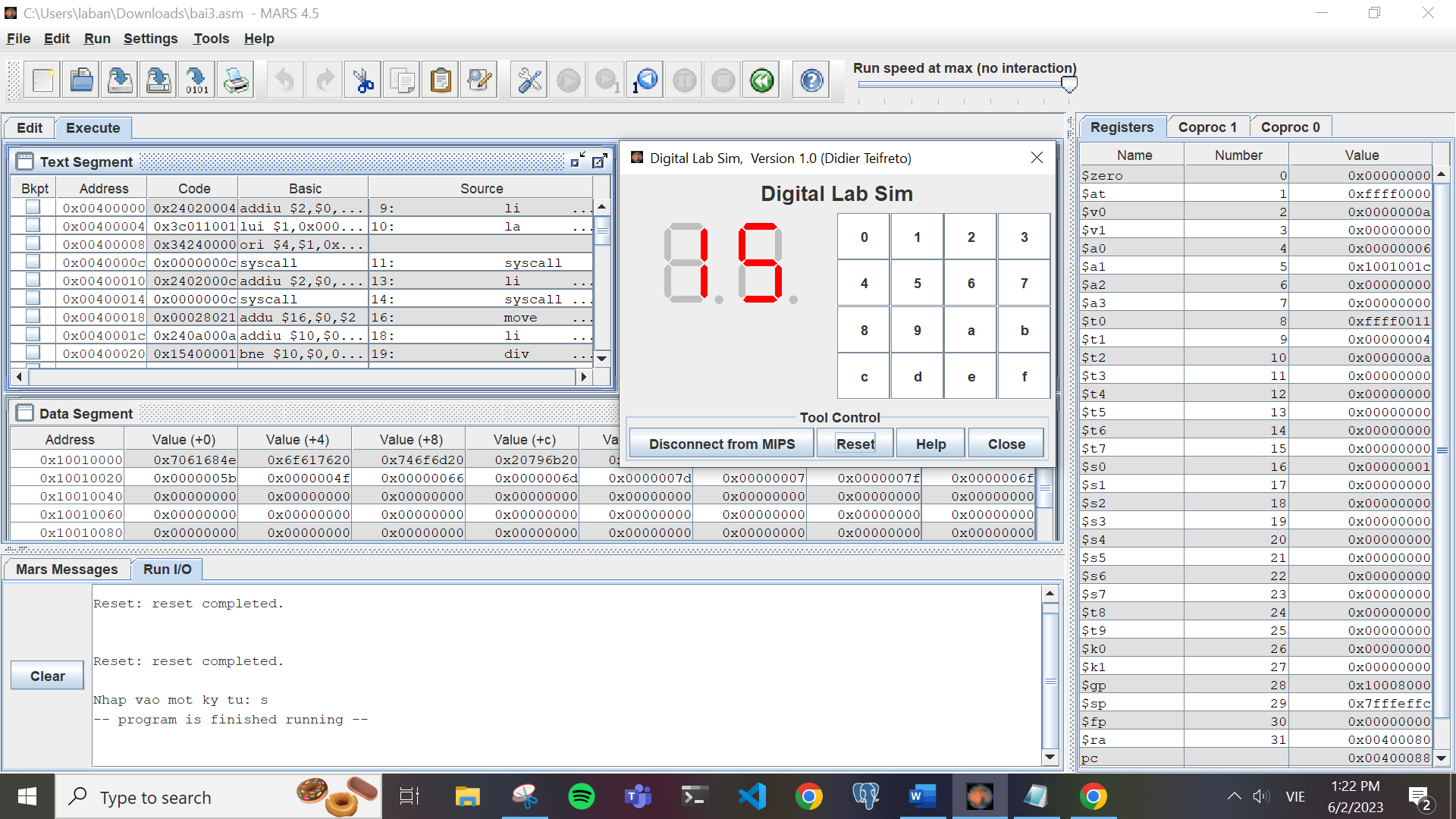
jr $ra

* Kết quả chạy thử:

Nhập ký tự A có mã là 65 và s có mã là 115

A screenshot of a computer

Description automatically generated



**Assignment 4:**

* Code:

.eqv MONITOR\_SCREEN 0x10010000

.eqv RED 0x00FF0000

.eqv GREEN 0x0000FF00

.eqv BLUE 0x000000FF

.eqv WHITE 0x00FFFFFF

.eqv YELLOW 0x00FFFF00

.text

li $k0, MONITOR\_SCREEN

li $t2, 2 # cac khoi tao

li $t4, 8

li $t1, -1

li $t4, 0

li $s1, 4

add $k0, $k0, -4

FOR:

addi $t1, $t1, 1 # i = 0, i ++

addi $t4, $t4, 1 # j++

beq $t1, 72, EXIT # i = 72 stop

add $k0, $k0, $s1 # $k0 += 4

div $t1, $t2 # i / 2

mfhi $t3 # $t3 = i % 2

bne $t4, 8, continue # j = 8 => i++

li $t4, 0 # j = 0

addi $t1, $t1, 1 # i++

continue:

beq $t3, $zero, doi\_mau

li $t0, WHITE

sw $t0, 0($k0)

nop

j FOR

doi\_mau:

li $t0, BLUE

sw $t0, 0($k0)

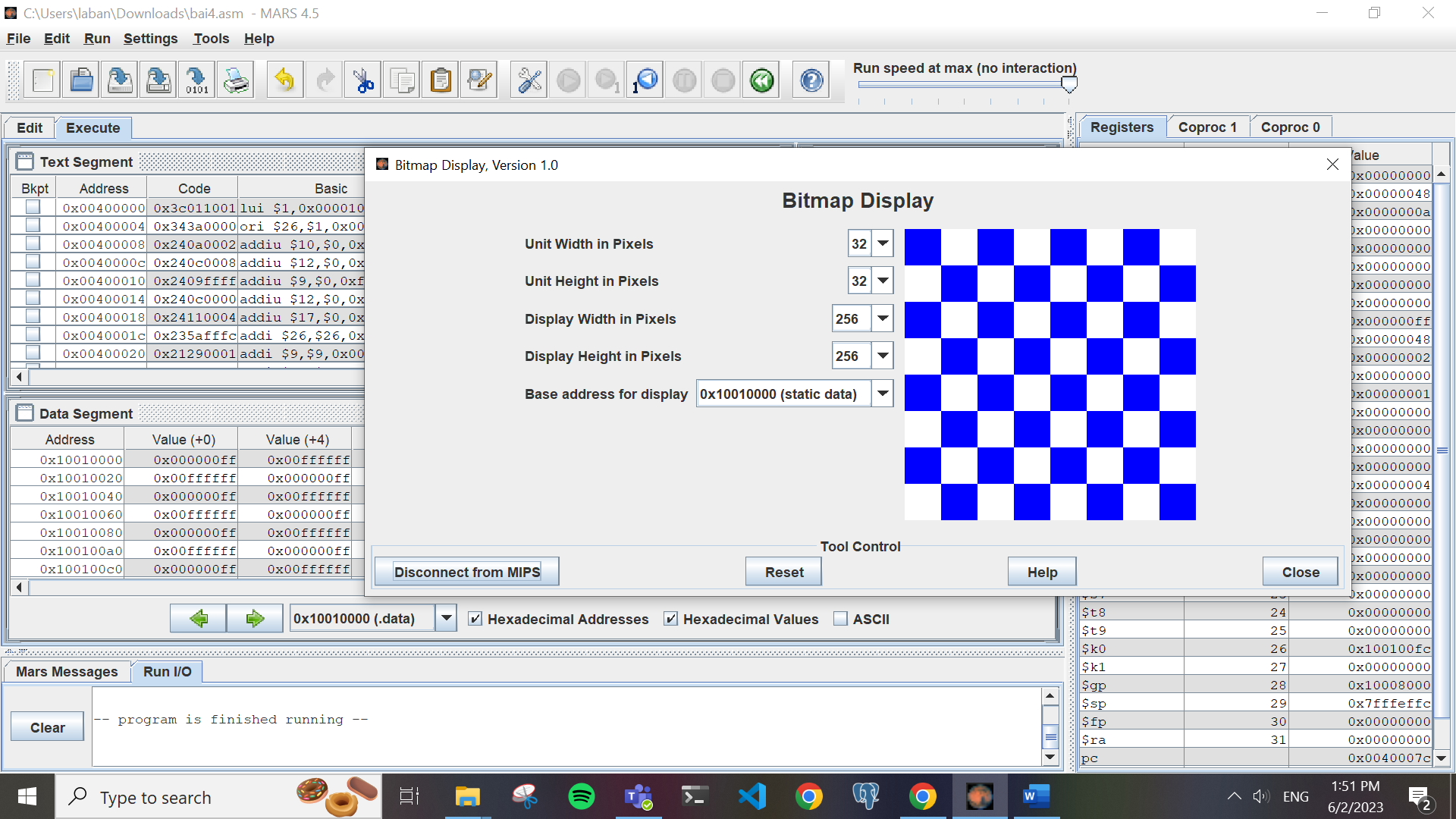
nop

j FOR

EXIT: li $v0, 10

syscall

* Kết quả chạy thử:



**Assignment 5:**

* Code:

.eqv MONITOR\_SCREEN 0x10010000

.eqv RED 0x00FF0000

.eqv GREEN 0x0000FF00

.data

x1: .asciiz "Nhap x1: "

y1: .asciiz "Nhap y1: "

x2: .asciiz "Nhap x2: "

y2: .asciiz "Nhap y2: "

error1: .asciiz "x2 phai khac x1\n"

error2: .asciiz "y2 phai khac y1\n"

.text

li $k0, MONITOR\_SCREEN

li $v0, 4

la $a0, x1

syscall

li $v0, 5

syscall

move $s0, $v0

li $v0, 4

la $a0, y1

syscall

li $v0, 5

syscall

move $s1, $v0

NhapX2:

li $v0, 4

la $a0, x2

syscall

li $v0, 5

syscall

move $s2, $v0

beq $s2, $s0, Error1

NhapY2:

li $v0, 4

la $a0, y2

syscall

li $v0, 5

syscall

move $s3, $v0

beq $s3, $s1, Error2

j continue

Error1:

li $v0, 4

la $a0, error1

syscall

j NhapX2

Error2:

li $v0, 4

la $a0, error2

syscall

j NhapY2

continue:

slt $t0, $s0, $s2

slt $t1, $s1, $s3

beq $t0, 0, Case3

beq $t1, 0, Case2

Case1: add $v0, $s1, $zero

loop\_1:

bgt $v0, $s3, Exit

add $v1, $s0, $zero

loop\_2:

bgt $v1, $s2, endloop\_2

beq $v0, $s1, InVien1

beq $v0, $s3, InVien1

beq $v1, $s0, InVien1

beq $v1, $s2, InVien1

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, GREEN

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j loop\_2

InVien1:

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, RED

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j loop\_2

endloop\_2:

add $v0, $v0, 1

j loop\_1

Case2:

add $v0, $s3, $zero

loop\_3:

bgt $v0, $s1, Exit

add $v1, $s0, $zero

loop\_4:

bgt $v1, $s2, endloop\_4

beq $v0, $s1, InVien2

beq $v0, $s3, InVien2

beq $v1, $s0, InVien2

beq $v1, $s2, InVien2

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, GREEN

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j loop\_4

InVien2:

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, RED

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j loop\_4

endloop\_4:

add $v0, $v0, 1

j loop\_3

Case3:

beq $t1, 0, Case4

add $v0, $s1, $zero

loop\_5:

bgt $v0, $s3, Exit

add $v1, $s2, $zero

loop\_6:

bgt $v1, $s0, endloop\_6

beq $v0, $s1, InVien3

beq $v0, $s3, InVien3

beq $v1, $s0, InVien3

beq $v1, $s2, InVien3

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, GREEN

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j loop\_6

InVien3:

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, RED

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j loop\_6

endloop\_6:

add $v0, $v0, 1

j loop\_5

Case4: add $v0, $s3, $zero

loop\_7:

bgt $v0, $s1, Exit

add $v1, $s2, $zero

loop\_8:

bgt $v1, $s0, endloop\_8

beq $v0, $s1, InVien4

beq $v0, $s3, InVien4

beq $v1, $s0, InVien4

beq $v1, $s2, InVien4

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, GREEN

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j loop\_8

InVien4:

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, RED

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j loop\_8

endloop\_8:

add $v0, $v0, 1

j loop\_7

Exit: li $v0, 10

syscall

* Kết quả chạy thử:

Bộ số (3,27) và (27,3)

