

Báo cáo thực hành giữa kì

Họ và tên : Đặng Quang Đạt

MSSV : 20205064

Phần A câu 3

.data

N: .asciiz "Nhap N: "

Co: .asciiz "Co "

Khong: .asciiz "Khong "

Trung: .asciiz "trung nhau"

Khongdung: .asciiz "hop le"

.text

j main

Cotrung:

li \$v0, 59

la \$a0, Co

la \$a1, Trung

syscall

j exit

Khongtrung:

li \$v0, 59

la \$a0, Khong

la \$a1, Trung

syscall

j exit

Khonghople:

```

li    $v0, 59
la    $a0, Khong
la    $a1, Khongdung
syscall
j      exit

```

main:

```

#Nhap N
li    $v0, 51
la    $a0, N
syscall

slti   $a3, $a0, 10    #kiem tra du lieu vao
bne    $a3, $0, Khonghople
move   $s1, $a0        #s1 = N
li     $t1, 0          #khoi chay so chu so la 0
li     $t2, 10         #hang so 10
li     $t4, 0          #bien chay vong lap

```

#Vong lap tim so chu so cua N

loop:

```

div    $s1, $t2
addi   $t1, $t1, 1
mfhi   $t0          #lay phan du phep chia cho vao stack
addi   $sp, $sp, -4
sw     $t0, 0($sp)
mflo   $s1
beq    $s1, $0, kiemtra    #Khi phan nguyen = 0 thi ket thuc vong lap
j      loop

```

kiemtra:

```
addi    $t4, $t4, 1
beq     $t4, $t1, Khongtrung #bien dem
lw      $s2, 0($sp)          #lay du lieu tu stack ra
move    $t9, $sp              #luu dia chi con tro hien tai vao cho moi
sub     $t5, $t1, $t4         #t5 so lan chay vong lap con
li      $t6, 0                #bien dem vong lap con
```

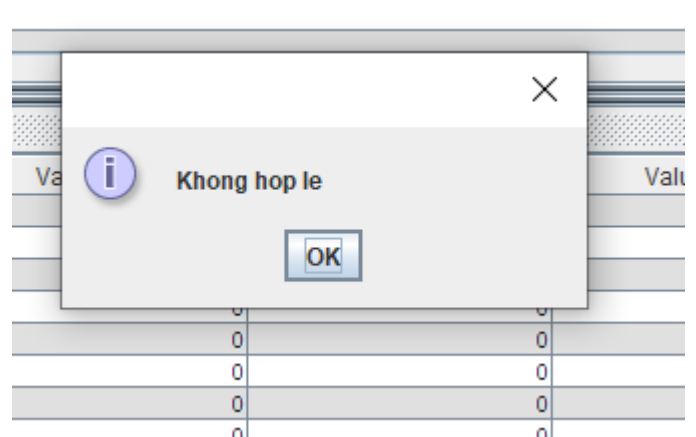
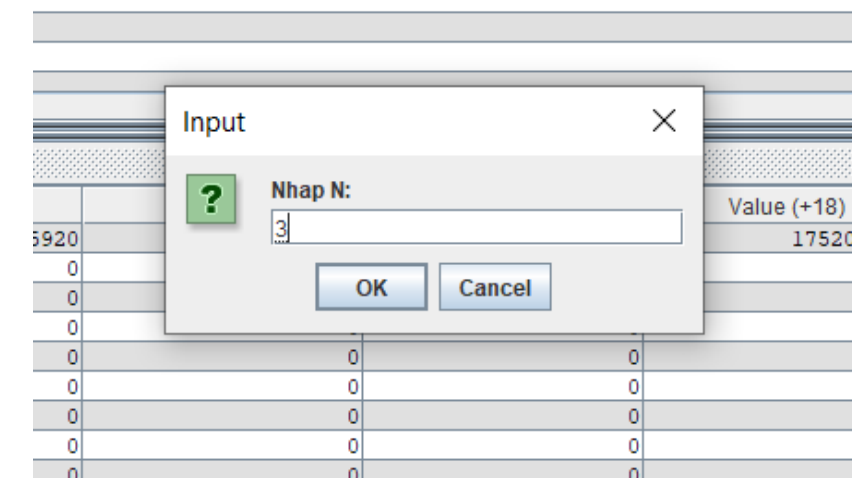
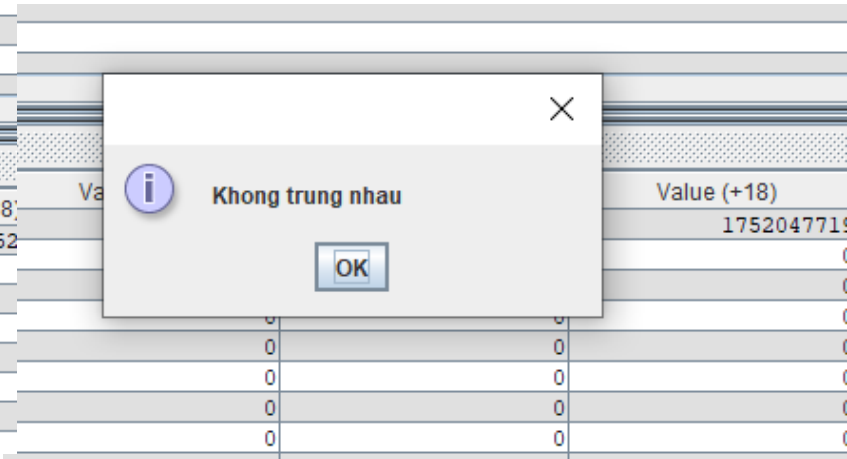
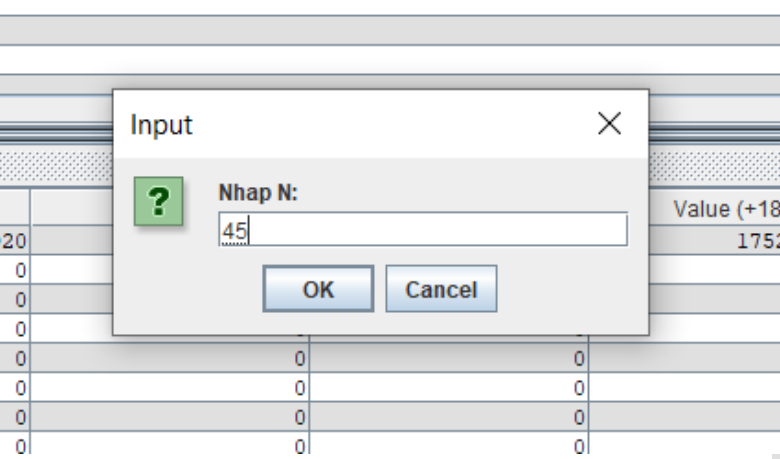
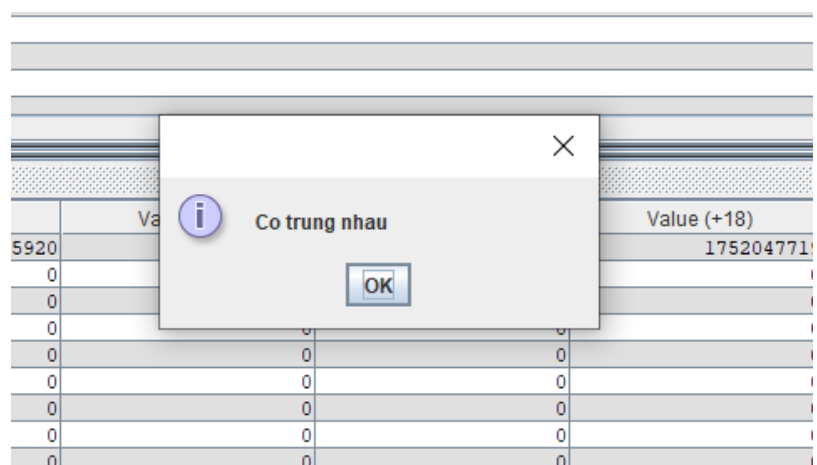
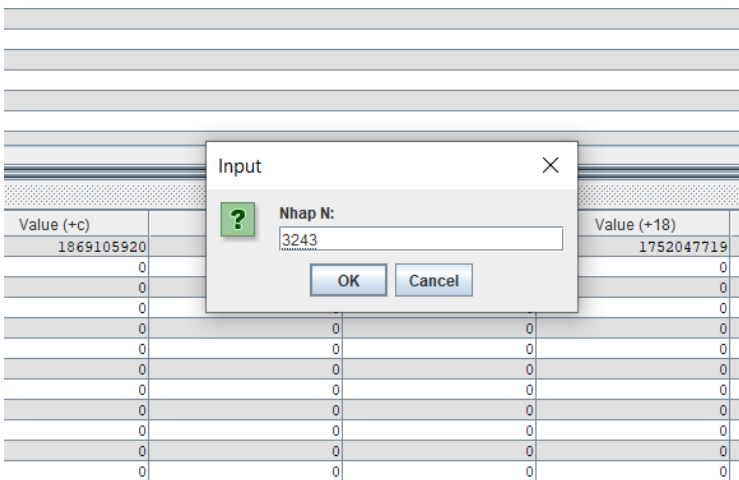
vonglap:

```
addi    $t6, $t6, 1
addi    $t9, $t9, 4
lw      $s3, 0($t9)
beq     $s3, $s2, Cotrung
beq     $t6, $t5, a           #khi t6 = t5 thi dung vong lap con
j       vonglap
```

a:

```
addi    $sp, $sp, 4
j       kiemtra
```

exit:



Cách thực hiện:

1. Nhập N, kiểm tra điều kiện N 2 chữ số
2. Chạy vòng lặp để tìm số chữ số của N
3. Chia N cho 10, lấy phần dư lưu vào stack
4. Lấy từng số của stack so sánh với các số còn lại

Bài 2 câu 5

.data

msg_input: .ascii "Nhap So Luong Phan Tu Mang: "

msg_input2: .ascii "Nhap Phan Tu: "

msg_input3: .ascii "Nhap M: "

space: .ascii "\t"

msg_output: .ascii "Ket Qua: "

arr: .word 0:100

.text

main:

Nhập So Luong Phan Tu

la \$a0,msg_input

li \$v0,4

syscall

Chuyển qua thanh ghi \$t0

li \$v0,5

syscall

move \$t0,\$v0 # t0: so phan tu cua mang

Nhập các phần tử của mảng

li \$v0,4

la \$a0,msg_input2

syscall

li \$t1,0

la \$t2,arr

move \$t6, \$t2

lap:

li \$v0,5

syscall

sw \$v0,(\$t2)

addi \$t1,\$t1,1

```

        addi $t2,$t2,4
        blt $t1,$t0,lap
li $t1,0
la $t2,arr
la $t5, arr
addi $t0, $t0, 1
# Nhap M
        li $v0,4
        la $a0,msg_input3
        syscall
        li $v0,5
        syscall
        move $t3,$v0          #t3: M
#xu li
        mul    $t5, $t0, 4
        add    $t2, $t2, $t5
        addi   $t2, $t2, -4
        sw     $t3, ($t2)
loop:

        lw     $s1, ($t2)
        addi   $t9, $t2, -4
        lw     $s2, ($t9)
        slt    $s3, $s2, $s1
        bne    $s3, $0, a
        j      inmang
a:
        sw     $s1, ($t9)
        sw     $s2, ($t2)
b:
        addi   $t2, $t2, -4
        beq    $t2, $t6, inmang
        j      loop
li $t1,0
la $t2,arr
#in mang
inmang:
        lw     $t4, ($t6)
        li     $v0, 1
        move   $a0, $t4
        syscall

```

```

# KHoang Trang
li $v0,4
la $a0,space
syscall
addi $t1,$t1, 1
addi $t6,$t6, 4
beq $t0,$t1, exit
j      inmang

```

exit:

The screenshot shows the Mars MIPS simulator interface. The 'Run I/O' tab is selected, and the output window displays the following text:

```

Nhap So Luong Phan Tu Mang: 4
Nhap Phan Tu: 43
30
20
14
Nhap M: 25
43      30      25      20      14
-- program is finished running (dropped off bottom) --

```

On the left side of the output window, there is a 'Clear' button.

Câu 6 Phần C

```

.data
nhap: .asciiz "Nhap xau : "
str: .asciiz " Ch"
space: .ascii " "
xau: .space 100
sotu: .asciiz "so tu bat dau bang Ch la: "
.text
main:

```

```

li      $v0,4
la      $a0, nhap
syscall
#Nhap xau tu ban phim
la      $a0, xau
li      $a1, 100
li      $v0, 8
syscall

```

la \$t6, xau

la \$t7, str

```
li    $a1, 0    # So tu Ch can tinh
addi  $t6, $t6, -1 #Them space cho tu dau tien
lb    $s0, 0($t7) # s0 = " "
lb    $s1, 1($t7) # s1 = "C"
lb    $s2, 2($t7) # s2 = "h"
```

loop:

#Doc lan luot 3 chu cai lien nhau trong xau

```
lb    $t1, 0($t6)
lb    $t2, 1($t6)
lb    $t3, 2($t6)
beq   $t2, $0, exit    #Chu cai thu 2 = 0 thi exit
addi  $t6, $t6, 1      #Nhay sang chu cai tiep
#Kiem tra 3 chu cai co trung " Ch" khong
bne   $s0, $t1, loop
bne   $s1, $t2, loop
bne   $s2, $t3, loop
add   $a1, $a1, 1      #Trung thi bien dem a1 + 1
j     loop
```

exit:

```
li    $v0, 56
la    $a0, sotu
syscall
```

The screenshot shows a debugger window with two main panes. The top pane displays assembly code with columns for Displacement (Bt), Address, Code, Basic, and Source. The bottom pane shows the Data Segment with columns for Address, Value (+0), Value (+4), Value (+8), Value (+C), and Value (+10). A dialog box is open over the Data Segment, displaying a table of values.

Address	Value (+0)	Value (+4)	Value (+8)	Value (+C)	Value (+10)
0x10010000	185431894	1869322016	2112032	6532928	0
0x10010020	0	0	0	0	0
0x10010040	0	0	0	0	0
0x10010060	0	0	0	0	0
0x10010080	1446294369	543649377	1814063171	2112097	0
0x100100A0	0	0	0	0	0
0x100100C0	0	0	0	0	0
0x100100E0	0	0	0	0	0
0x10010100	0	0	0	0	0
0x10010120	0	0	0	0	0
0x10010140	0	0	0	0	0
0x10010160	0	0	0	0	0
0x10010180	0	0	0	0	0
0x100101A0	0	0	0	0	0

The dialog box is titled "so tu bat dau bang Ch la: 2" and has an "OK" button.