

## Bài 1 (Câu 6): Nhập 3 số nguyên a, b, c, kiểm tra đây có phải là 3 cạnh của một tam giác không?

- Cách thực hiện:
  - B1: Nhập lần lượt 3 số nguyên và kiểm tra 3 số nguyên có lớn hơn 0 hay không. Nếu bé hơn bằng 0 in ra màn hình kết quả và kết thúc. Nếu lớn hơn 0 thì tiếp tục
  - B2: Lần lượt kiểm tra tổng 2 cạnh có lớn hơn cạnh còn lại hay không nếu không thì in ra màn hình và kết thúc chương trình nếu có tiếp tục kiểm tra các trường hợp còn lại.
  - B3: Nếu kiểm tra đủ và đúng 3 trường hợp ( $a+b > c$ ,  $b+c > a$ ,  $a+c > b$ ) thì in ra màn hình và kết thúc chương trình
- Kết quả

```
-- program is finished running --  
  
Nhập vào số nguyên a: -1  
Đây là không phải ba cạnh một tam giác:  
-- program is finished running --
```

```
Nhập vào số nguyên a: 3  
Nhập vào số nguyên b: 4  
Nhập vào số nguyên c: 5  
Đây là ba cạnh một tam giác  
-- program is finished running --
```

```
Nhập vào số nguyên a: 1  
Nhập vào số nguyên b: 1  
Nhập vào số nguyên c: 2  
Đây là không phải ba cạnh một tam giác:  
-- program is finished running --
```

- Mã nguồn

```
20184113_NguyenVanHong_Giucky_Bai2.asm 20184113_NguyenVanHong_Giucky_Bai3.asm mips5.asm  
mips1.asm 20184113_NguyenVanHong_Giucky_Bai1.asm  
1  
2 .data  
3 str1: .asciz "Nhập vào số nguyên a: "  
4 str2: .asciz "Nhập vào số nguyên b: "  
5 str3: .asciz "Nhập vào số nguyên c: "  
6 str4: .asciz "Đây là ba cạnh một tam giác "  
7 str5: .asciz "Đây là không phải ba cạnh một tam giác: "  
8  
9 .text  
10  
11 #-----nhập cạnh a-----  
12 # In chuỗi "Nhập vào số nguyên a: "  
13 li $v0, 4 # $v0 -> 4  
14 la $a0, str1 # $a0 -> address(str1)  
15 syscall  
16 # Nhập giá trị  
17 li $v0, 5 # $v0 -> 5  
18 syscall  
19 move $s1, $v0 # $s1 -> $v0  
20 # kiểm tra giá trị > 0  
21 blez $s1, KhôngPhaiTamGiac # $s1 <= 0 true nhảy đến nhận KhôngPhaiTamGiac  
22  
23 #-----nhập cạnh b-----  
24 # In chuỗi "Nhập vào số nguyên a: "  
25 li $v0, 4 # $v0 -> 4  
26 la $a0, str2 # $a0 -> address(str2)
```

```

Edit  Execute
20184113_NguyenVanHong_Giuary_Bai2.asm  20184113_NguyenVanHong_Giuary_Bai3.asm*  mips5.asm
mips1.asm  20184113_NguyenVanHong_Giuary_Bai1.asm
25 li $v0, 4           #$v0 -> 4
26 la $a0, str2        #$a0 -> address(str2)
27 syscall
28 # Nhap gia tri
29 li $v0, 5           #$v0 -> 5
30 syscall
31 move $s2, $v0        # $s2 -> $v0
32 # kiem tra gia tri > 0
33 blez $s2, KhongPhaiTamGiac  # $s2 <= 0 true chuyen den  KhongPhaiTamGiac
34
35 #-----nhap canh c-----
36 # In chuỗi "Nhap vao so nguyen a: "
37 li $v0, 4           #$v0 -> 4
38 la $a0, str3        #$a0 -> address(str3)
39 syscall
40 # Nhap gia tri
41 li $v0, 5           #$v0 -> 5
42 syscall
43 move $s3, $v0        # $s3 -> $v0
44 # kiem tra gia tri > 0
45 blez $s3, KhongPhaiTamGiac  # $s3 <= 0 true nhay den nhan KhongPhaiTamGiac
46
47
48 #xet th a+b>c
49 add $s0, $s1, $s2    # $s1 + $s2 -> $s0
50 sgt $t0, $s0, $s3    # so sánh $s0 vs $s3 -> $t0

```

```

Edit  Execute
20184113_NguyenVanHong_Giuary_Bai2.asm  20184113_NguyenVanHong_Giuary_Bai3.asm*  mips5.asm
mips1.asm  20184113_NguyenVanHong_Giuary_Bai1.asm
43 move $s3, $v0        # $s3 -> $v0
44 # kiem tra gia tri > 0
45 blez $s3, KhongPhaiTamGiac  # $s3 <= 0 true nhay den nhan KhongPhaiTamGiac
46
47
48 #xet th a+b>c
49 add $s0, $s1, $s2    # $s1 + $s2 -> $s0
50 sgt $t0, $s0, $s3    # so sánh $s0 vs $s3 -> $t0
51 bne $t0, 1, KhongPhaiTamGiac  # $t0 != 1 true chuyển đến KhongPhaiTamGiac
52
53 #xet th b+c>a
54 add $s0, $s2, $s3    # $s2 + $s3 -> $s0
55 sgt $t0, $s0, $s1    # so sánh $s0 vs $s3 -> $t0
56 bne $t0, 1, KhongPhaiTamGiac  # $t0 != 1 true chuyển đến KhongPhaiTamGiac
57
58 #xet th a+c>b
59 add $s0, $s1, $s3    # $s1 + $s3 -> $s0
60 sgt $t0, $s0, $s2    # so sánh $s0 vs $s3 -> $t0
61 bne $t0, 1, KhongPhaiTamGiac  # $t0 != 1 true chuyển đến KhongPhaiTamGiac
62
63 #-----In chuỗi "Day la ba canh mot tam giac: "-----
64 li $v0, 4           #$v0 -> 4
65 la $a0, str4        #$a0 -> address(str5)
66 syscall
67 #chuyển đến label fin
68

```

Bài 2(Câu 9): Nhập mảng số nguyên từ bàn phím. In ra màn hình số chẵn nhỏ nhất lớn hơn mọi số lẻ trong mảng.

Bài 3(Câu 3): Nhập vào chuỗi ký tự. Đổi các ký tự chữ hoa thành chữ thường và chữ thường thành chữ hoa

- Giải thích
- Kết quả

```

-- program Simulated MIPS console input and output
Nhap vao chuỗi: NguyenVanHong
Chuoi sau khi chuyen: nGUYENVANHONG*
-- program is finished running --

```

- Code

```

.data
string: .asciiz
str1: .asciiz "Nhap vao chuoi: "
str2: .asciiz "Chuoi sau khi chuyen: "
.text
.globl main

main:

# in chuoi "Nhap vao chuoi: " ra man hinh
li $v0, 4      # $v0 -> 4
la $a0, str1   # $a0-> address(str1)
syscall

# Nhap vao chuoi
li $v0, 8      # $v0 -> 4
la $a0, string # $a0-> address(str1)
li $a1, 100    # $a1->10
syscall

    la $t0, string
    move $t1, $0
    add $t2, $0, 0x20 #cac chu hoa chu thuong hon nhau 32
    add $t3, $0, 0x5B #lon hon 5b la chu thuong

```

loopProc:

```

25
26 loopProc:
27     lb $t1, ($t0)
28     nop
29     beq $t1, $0, doneProc
30     nop
31     move $t4, $t3
32     sub $t5, $t1, $t4
33     bge $t5, $0, helperProc #check chu thuong hay chu hoa
34     add $t1, $t1, $t2 #chuyen chu hoa thanh chu thuong
35     sb $t1, ($t0)
36     add $t0, $t0, 0x01
37
38     nop
39     j loopProc
40 helperProc:
41     nop
42     sub $t1, $t1, $t2 #chuyen chu thuong thanh chu hoa
43     sb $t1, ($t0)
44     add $t0, $t0, 0x01
45
46     nop
47     j loopProc
48
49 doneProc:
50 # in chuoi "Nhap vao chuoi: " ra man hinh

```

```

36     add $t0, $t0, 0x01
37
38     nop
39     j loopProc
40 helperProc:
41     nop
42     sub $t1, $t1, $t2 #chuyen chu thuong thanh chu hoa
43     sb $t1, ($t0)
44     add $t0, $t0, 0x01
45
46     nop
47     j loopProc
48
49 doneProc:
50 # in chuoi "Nhap vao chuoi: " ra man hinh
51 li $v0, 4      # $v0 -> 4
52 la $a0, str2   # $a0-> address(str1)
53 syscall
54
55 la $a0, string
56 add $t0, $0, $0
57 li $v0, 4 #in ki tu ra
58 syscall
59 li $v0, 10
60 syscall
61

```