

BÁO CÁO BÀI THỰC HÀNH GIỮA KÌ

Họ và Tên : Nguyễn Kim Bảo
MSSV : 20194486

A.Số nguyên

Code:

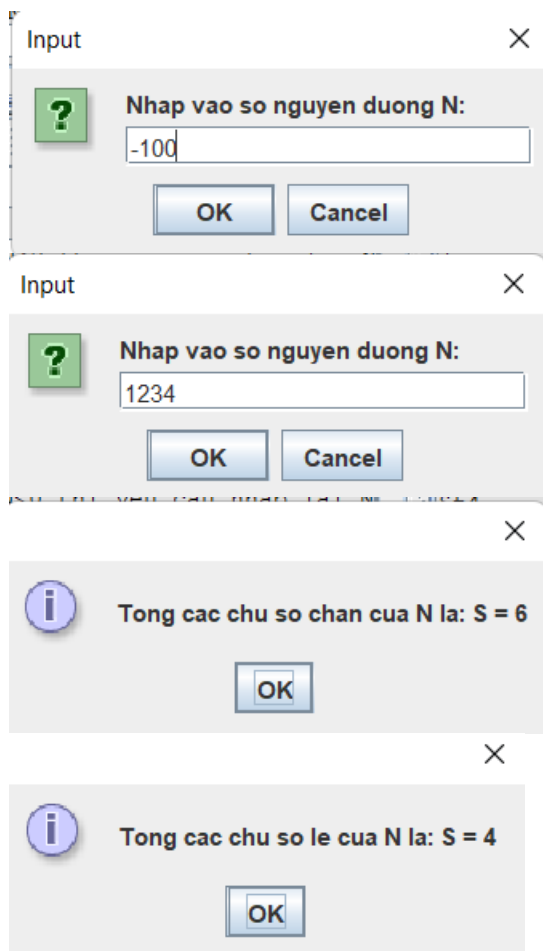
```
1  .data
2  Message_Input : .ascii "Nhap vao so nguyen duong N: "
3  Message_OutputEven: .ascii "Tong cac chu so chan cua N la: S = "
4  Message_OutputOdd: .ascii "Tong cac chu so le cua N la: S = "
5  .text
6  main:
7      li $v0, 51
8      la $a0, Message_Input
9      syscall
10     add $t1,$a0,0 #Gan gia tri N cho t1
11     li $t2, 10 #Gan gia tri thanh ghi t2 = 10
12     li $t3, 2 #Gan gia tri thanh ghi t3 = 2
13     li $t4, 0 #Gan gia tri ban dau cua tong cac chu so chan = 0
14     li $t5, 0 #Gan gia tri ban dau cua tong cac chu so le = 0
15     li $t6, 1 #Gan gia tri thanh ghi t6 =1
16     blt $t1, $t6, main #Neu N<0 thi yeu cau nhap lai N
17
18  div_10:
19     div $t1, $t2 #Chia so nguyen co dau
20     mflo $a1 # Thuong
21     mfhi $a2 # Phan du
22     div $a2, $t3 #Chia so nguyen co dau
23     mfhi $a3 #Phan du
24     beq $a3 $zero, even #Neu chia het thi chuyen den doan lenh even
```

```

25 odd:
26     add $t5, $t5, $a2 # t5 += a2
27     j check #Nhay den doan lenh check
28 even:
29     add $t4, $t4, $a2 # t4 += a2
30     j check #Nhay den doan lenh check
31 check:
32     beq $a1, $zero, done #Thuong = 0 -> done
33     add $t1, $a1, 0 # t1 = a1
34     j div_10 #Nhay den doan lenh div_10
35
36 done:
37     li $v0, 56
38     la $a0, Message_OutputEven
39     add $a1, $t4, $zero
40     syscall
41     li $v0, 56
42     la $a0, Message_OutputOdd
43     add $a1, $t5, $zero
44     syscall

```

Kết quả :



Phân tích cách thực hiện :

- Yêu cầu nhập vào số nguyên dương N. Nếu $N < 0$ yêu cầu nhập lại.
- Thực hiện phép chia lấy phần nguyên và phần dư của số N nhập vào cho 10 .
- Tiếp tục thực hiện phép lấy phần dư của phần dư vừa thu được ở trên với 2 . Nếu chia hết cho 2 -> Số chẵn -> Cộng vào tổng của các chữ số chẵn của N. Ngược lại cộng vào tổng các chữ số lẻ của N

B. Mảng

Code:

```
1 .data
2     A: .word 5, 4, 2
3     n: .word 3 #So phan tu mang
4
5 .text
6     la $a0, n #Luu dia chi n tai a0
7     lw $a1, 0($a0)
8     addi $a2, $zero, 3 #luu M = 3 tai a2
9     la $a3, A #luu d/c A[0] tai a3
10
11     #nhap A[n] = m; n++
12     sll $s3, $a1, 2
13     add $s4, $s3, $a3
14     sw $a2, 0($s4)
15     add $a1, $a1, 1
16
17     #Bubble sort
18     main:
19     j bubble_sort #sort
20 after_sort:
21     li $v0, 10 #exit
22     syscall
23 end_main:
24
25 bubble_sort:
26     li $t0, 1 # isSwap? t0 = 1 -> van con swap
27     subi $a1, $a1, 1 # a1 = n-1
28     loop:
29         beq $t0, $zero, endloop # t0=0 -> k con swap -> end
30         li $t0, 0
31         add $t1, $zero, $zero # i = 0
32         j loop2
33     endloop:
34 end_bubble_sort:
35     j after_sort
36
37 loop2:
38     slt $t2, $t1, $a1 # i < n - 1 ??
39     beq $t2, $zero, end_loop2 # t2 = 0 -> i >= n -> false
40     sll $t3, $t1, 2 # t3 = 4*i
41
42     add $a2, $a3, $t3 # a2 = d/c A[i]
```

```

43      lw $t4 , 0($a2) #t4 = A[i]
44      lw $t5 , 4($a2) #t5 = A[i+1]
45
46      sge $t2, $t4, $t5 # t4 >= t5 ??
47      addi, $t1, $t1, 1 # i++
48
49      bne $t2, $zero, loop2 # t2 = 1 -> ko can swap
50      li $t0, 1 #cap nhac t0 = 1
51      sw $t4, 4($a2) #swap A[i] va A[i+1]
52      sw $t5, 0($a2)
53      j loop2
54 end_loop2:
55      j loop

```

Chương trình con

- Bubble_sort: để chạy sắp xếp nổi bọt.
 - + loop: vòng lặp break khi không còn thao tác swap
 - + loop2: for i =0 tới n-1, nếu A[i] nhỏ hơn A[i+1] thì swap

Kết quả :

Mảng ban đầu :

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	5	4	2	3	0	0	0	0
0x10010020	0	0	0	0	0	0	0	0
0x10010040	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0
0x100100e0	0	0	0	0	0	0	0	0
0x10010100	0	0	0	0	0	0	0	0
0x10010120	0	0	0	0	0	0	0	0

Mảng sau khi kết thúc chương trình :

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	5	4	3	2	0	0	0	0
0x10010020	0	0	0	0	0	0	0	0
0x10010040	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0
0x100100e0	0	0	0	0	0	0	0	0
0x10010100	0	0	0	0	0	0	0	0
0x10010120	0	0	0	0	0	0	0	0

C.Xâu ký tự

Code:

```
1  .data
2      input:  .space 201
3      string: .asciiz "So tu trong xau bat dau bang 'Ch' la: "
4      check1: .byte 'C'
5      check2: .byte 'h'
6  .text
7      main:
8          la $a0, input
9          li $a1, 201  #Gioi han chi duoc nhap 200 ky tu
10         li $v0, 8     #Do va luu vao $a0
11         syscall
12         nop
13
14         jal coutCh  #jump and link
15
16         li $v0, 56
17         la $a0, string
18         add $a1, $t3, $zero #Gan t3 = a1
19         syscall
20         nop
21
22         li $v0, 10
23         syscall
24         nop
25
26     coutCh:
27         la $s0, input #Luu lai vao s0
28
29         #Lay ra 2 ky tu C va h
30         la $s1, check1
31         la $s2, check2
32
33         lb $t1, 0($s1)
34         lb $t2, 0($s2)
35
36         li $t3, 0 # So tu bat dau bang 'Ch'
37         subi $s0, $s0, 1 # Tam thoi tru 1 de vong lap co the chi den ky tu dau tien
38
39     loop:
40         addi $s0, $s0, 1
41         lb $t0, 0($s0)
42         beq $t0, $zero, endLoop
43     compareC:
44         bne $t0, $t1, skip # Neu ky tu dau tien khong bang voi ky tu 'C' thi nhay den skip de den tu tiep theo
45     compareh:
46         addi $s0, $s0, 1
47         lb $t0, 0($s0)
48         beq $t0, $zero, endLoop
49         bne $t0, $t2, skip # Neu ky tu tiep theo khong bang voi ky tu 'h' thi nhay den skip de den tu tiep theo
50         addi $t3, $t3, 1
```

```

50      skip:
51          addi $s0, $s0, 1
52          lb $t0, 0($s0)
53          beq $t0, $zero, endLoop
54          bne $t0, 32, skip
55      j loop
56  endLoop:
57  endCoutCh:
58      jr $ra
59
60

```

Kết quả:

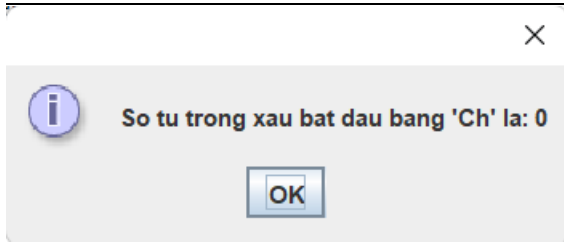
Text Segment					Labels	
Bkpt	Address	Code	Basic	Source	Label	Address
	0x00400000	0x3c011001	lui \$1,4097	8: la \$a0,input	main	0x00400000
	0x00400004	0x34240000	ori \$4,\$1,0		coutCh	0x00400040
	0x00400008	0x240500c9	addiu \$5,\$0,201	9: li \$a1,201 #Gioi han chi duoc nhap 200 ..	loop	0x0040006c
	0x0040000c	0x24020008	addiu \$2,\$0,8	10: li \$v0,8 #Do va luu vao \$a0	compareC	0x00400078
	0x00400010	0x0000000c	syscall	11: syscall	compareh	0x0040007c
	0x00400014	0x00000000	nop	12: nop	skip	0x00400090
	0x00400018	0x0c100010	jal 0x00400040	14: jal coutCh #jump and link	endLoop	0x004000a8
	0x0040001c	0x24020038	addiu \$2,\$0,56	16: li \$v0, 56	endCoutCh	0x004000a8
	0x00400020	0x3c011001	lui \$1,4097	17: la \$a0, string	input	0x10010000
	0x00400024	0x342400c9	ori \$4,\$1,201		string	0x100100c9
	0x00400028	0x01602820	add \$5,\$11,\$0	18: add \$a1, \$t3, \$zero #Gan t3 = a1		
	0x0040002c	0x0000000c	syscall	19: syscall		

Data Segment								
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0	0	0	0	0	0	0	0
0x10010020	0	0	0	0	0	0	0	0
0x10010040	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0
0x100100e0	1851875872	1126637671	544166656	1948284276	1735290738	1969322016	1952539168	1969316896
0x10010100	0	0	0	2112097	26691	0	0	0
0x10010120	0	0	0	0	0	0	0	0

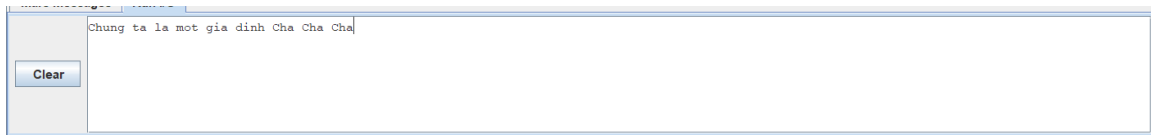
TH1: Không có ký tự 'Ch' trong chuỗi

thuc hanh kien truc may tinh

Clear

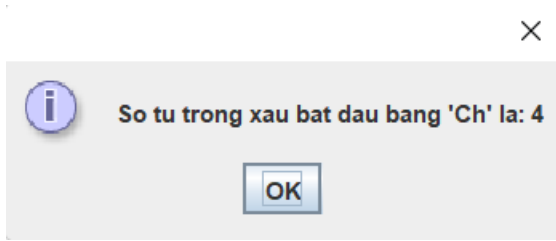


TH2: Có ký tự 'Ch' trong chuỗi



Chúng ta là một gia đình Cha Cha Cha

Clear



Phân tích cách thực hiện :

- Vòng lặp bắt đầu bằng việc đọc ký tự đầu tiên của 1 từ
- Nếu nó là ký tự kết thúc thì kết thúc vòng lặp
- Nếu không thì sẽ so sánh ký tự đầu tiên đó với ký tự 'C'
- Nếu sai, sẽ cho chạy hết từ đó, tức duyệt chuỗi cho đến khi gặp dấu cách (32) thì sẽ quay về vòng lặp
- Nếu đúng thì tiếp tục so sánh ký tự 2 với ký tự 'h'
- Nếu sai, sẽ cho chạy hết từ đó, tức duyệt chuỗi cho đến khi gặp dấu cách (32) thì sẽ quay về vòng lặp
- Nếu đúng thì cộng số lượng từ bắt đầu bằng 'Ch' thêm 1
- Quay về vòng lặp