Báo cáo bài giữa kì Môn thực hành kiến trúc máy tính

Phùng Trung Kiên MSSV: 20204994 Lớp Việt – Nhật 03 Trường đại học Bách Khoa Hà Nội,

Email: kien.pt204994@sis.hust.edu.vn

Đề bài:

- A.3: Nhập số nguyên dương N có từ 2 chữ số trở lên, kiểm tra N có chứa 2 chữ số trùng nhau không.
- **B.4:** Nhập mảng số nguyên từ bàn phím. In ra màn hình vị trí và giá trị của phần tử âm lớn nhất trong mảng.
- C.1: Nhập vào xâu ký tự. In ra màn hình từ ngắn nhất có trong xâu.

Cách thực hiện:

Bài làm:

syscall

- Mỗi sinh viên làm 3 bài thuộc 3 chủ đề, các bài được giao ngẫu nhiên.
- Yêu cầu cơ bản: thực hiện được các yêu cầu của đề bài. Yêu cầu nâng cao: xử lý được các trường hợp ngoại lệ. Kết quả thực hiện: Sinh viên soạn và nộp báo cáo, với mỗi bài cần trình bày:
- Phân tích cách thực hiện -

Ý nghĩa của các chương trình con nếu có

- Ảnh chụp màn hình kết quả thực hiện
- Mã nguồn, có chú thích Nộp báo cáo: Sinh viên đặt tên file báo cáo theo cú pháp MSSV_Hoten_MidTerm.pdf rồi nộp vào assignment

Dai iai			
A.3:			
Code:			
.data			
msg:		.asciiz	"Nhap so:"
tontai:		.asciiz	"Chu so do la: "
khongt	ontai:	.asciiz	"Khong ton tai chu so lap lai 2 lan"
.text			
main:			
li	\$v0, 51		
la	\$a0, msg		# In ra msg va cho so nhap vao a0

```
addi
       $t0, $a0, 0
                            # t0 = a0 + 0
li
       $a1, 10
                                   #a1 = 10
f1:
       #!!! ham f1 su dung de quy la f2 de duyet tat ca cac truong hop
                     # Neu t0 == 0 thi thoat
beqz $t0, no
div
       $t0, $a1
                            # Lay t0 / 10
mflo
       $t0
                            # Lay thuong (bo di hang don vi)
mfhi
       $t1
                            # So du khi chia 10
li
                            # t3 = 0
       $t3, 0
addi
       $s1, $t0, 0
                            # s1 = t0
f2:
beqz $s1, f1
                     # Neu s1 == 0 thi thoat
div
       $s1, $a1
                     # Lay s1 / 10
mflo
       $s1
                     # Lay thuong (bo di hang don vi)
mfhi
       $s2
                     # So du khi chia 10
       $s2, $t1, yes # If s2 == s1 => thoat
beq
j
       f2
                     # Neu khong thi tiep tuc de quy
yes:
li
       $v0, 56
       $a0, tontai
la
addi
       $a1, $s2, 0
syscall
       end
j
no:
li
       $v0, 55
       $a0, khongtontai
la
```

syscall

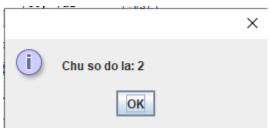
```
.data
msg:
              .asciiz
                              "Nhap so:"
                              "Chu so do la: "
tontai:
              .asciiz
khongtontai:
              .asciiz
                              "Khong ton tai chu so lap lai 2 lan"
.text
main:
       li
              $v0, 51
               $aO, msg
                              # In ra msg va cho so nhap vao a0
       la
       syscall
       addi
               $t0, $a0, 0
                                      # t0 = a0 + 0
                                      \# a1 = 10
               $al, 10
       1i
       f1:
               # !!! ham f1 su dung de quy la f2 de duyet tat ca cac truong hop
                     $t0, no
                                    # Neu t0 == 0 thi thoat
               beqz
                       $t0, $al
               div
                                             # Lay t0 / 10
               mflo
                       $t0
                                             # Lay thuong (bo di hang don vi)
                                             # So du khi chia 10
               mfhi
                       $t1
               1i
                       $t3, 0
                                              # t3 = 0
                       $s1, $t0, 0
                                              \# s1 = t0
               addi
               f2:
                       beqz
                              $s1, f1
                                              # Neu s1 == 0 thi thoat
                                              # Lay s1 / 10
                       div
                              $s1, $al
                       mflo
                                             # Lay thuong (bo di hang don vi)
                              $s1
                       mfhi
                              $s2
                                              # So du khi chia 10
                              $s2, $t1, yes # If s2 == s1 => thoat
                       beq
                                               # Neu khong thi tiep tuc de quy
                       j
yes:
               $v0, 56
       1i
        la
               $a0, tontai
               $al, $s2, 0
        addi
        syscall
               end
no:
              $v0, 55
       1i
       la
               $a0, khongtontai
        syscall
```

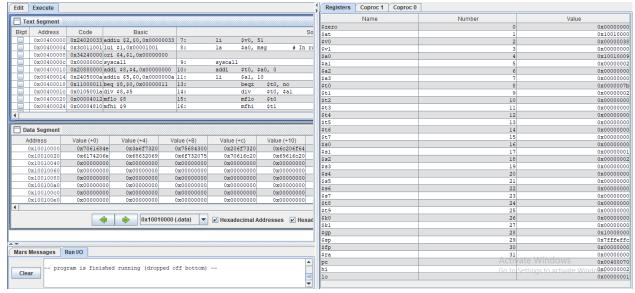
end:

Run:

1:



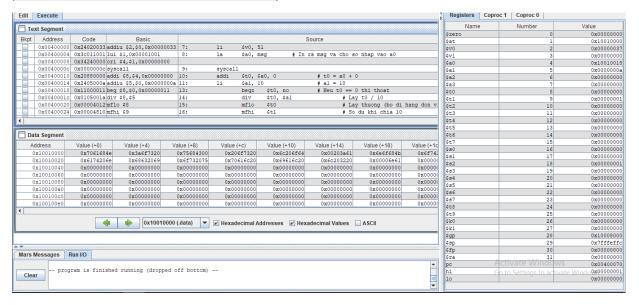




2:







Phân tích cách thực hiện:

Ví du ta có chữ số 93234

Đầu tiên tách số 4 ra được 9323 và 4 bằng cách chia 10 rồi lấy dư và thương

9323 4

Sau đó dùng đệ quy để liên tục tách các số còn lại, 3 => 2 => 3 => 9 đồng thời so sánh với 4, nếu một chữ số nào trong đó bằng 4 thì in ra 4

Nếu KHÔNG: ta chạy ngược lại 4 chữ số 9323 tiếp tục thực hiện tách và đệ quy như trên.

Đến khi còn lại 1 chữ số, thì ta xét thương của nó với 10 là 0 thì loại, hoặc đến khi tìm ra 2 chữ số giống nhau.

B4:

syscall

```
Code:
.data
msg1: .asciiz "Nhap phan tu thu "
msg2: .asciiz ": "
msg3: .asciiz "Nhap so phan tu cua mang: "
msg4: .asciiz "So am Ion nhat la "
msg5: .asciiz " o vi tri thu "
msg6: .asciiz "Mang khong co so am"
.text
main:
       $t0, $0, 0
add
                     # i
                     # in ra msg3
       $a0, msg3
la
li
       $v0, 4
syscall
li
       $v0, 5
syscall
add
       $s3, $0, -999999
                             #max
add
       $s1, $v0, $0
                     #n
sub
       $s4, $s1, 1
       $s5, $0, 0
add
                     # check
              $t0, $s1, end_input
f1:
       beq
la
       $a0, msg1
       $v0, 4
li
```

```
$a0, $t0, 1
add
li
       $v0, 1
syscall
la
       $a0, msg2
       $v0, 4
li
syscall
       $v0, 5
li
syscall
      $s2, $t0, 2
sll
      $s2, $s0, $s2
add
      $s0, $v0, $0
add
slt
       $a1, $s0, $0
      $a1, 1, f2
beq
      end_loop
j
f2:
              $s5, $0, 1
       add
      $a1, $s3, $s0
slt
      $a1, 0, end_loop
beq
add
       $s3, $s0, $0
add
       $t1, $t0, 1
                     # max_index
end_loop:
add
       $t0, $t0, 1
j
       f1
end_input:
      $s5, 0, end
beq
la
       $a0, msg4
li
       $v0, 4
syscall
li
       $v0, 1
add
       $a0, $0, $s3
syscall
```

la

li

\$a0, msg5

\$v0, 4

```
syscall
li
       $v0, 1
       $a0, $0, $t1
add
syscall
j end_all
              $a0, msg6
end: la
li
       $v0, 4
syscall
end_all
.data
        msgl: .asciiz "Nhap phan tu thu "
        msg2: .asciiz " : "
        msg3: .asciiz "Nhap so phan tu cua mang: "
        msg4: .asciiz "So am lon nhat la "
        msg5: .asciiz " o vi tri thu "
        msg6: .asciiz "Mang khong co so am"
 .text
main:
                add $t0, $0, 0
                                    # i
               la
                       $aO, msg3
                                     # in ra msg3
               li.
                       $v0, 4
                syscall
                       $v0, 5
               li.
                syscall
                      $s3, $0, -999999999999 #max
                add
                add
                       $s1, $v0, $0
               sub
                       $s4, $s1, 1
                    $s5, $O, O
                                      # check
               add
        f1:
                       $t0, $s1, end_input
               beq
                       $a0, msgl
               la
               li
                       $v0, 4
                syscall
                       $a0, $t0, 1
                add
```

```
$v0, 1
      1i
      syscall
            $a0, msg2
      la
      li
             $v0, 4
      syscall
            $v0, 5
      1i
      syscall
      sll
             $s2, $t0, 2
      add
             $s2, $s0, $s2
      add
             $s0, $v0, $0
      slt
             $al, $s0, $0
      beq
             $al, 1, f2
             end_loop
f2:
      add
             $s5, $0, 1
      slt
             $al, $s3, $s0
             $al, 0, end_loop
      beq
      add
             $s3, $s0, $0
      add
             $t1, $t0, 1
                          # max_index
end_loop:
            $t0, $t0, 1
      add
      j
end_input:
            $s5, 0, end
      beq
      la
            $aO, msg4
           li.
                   $v0, 4
           syscall
           li.
                    $v0, 1
           add
                    $a0, $0, $t1
           syscall
           j end_all
                  $aO, msg6
  end:
           la
           li
                     $v0, 4
           syscall
  end all:
```

Run:

1:

```
Nhap so phan tu cua mang: 5

Nhap phan tu thu 1 : 1

Nhap phan tu thu 2 : 2

Nhap phan tu thu 3 : 3

Nhap phan tu thu 4 : 4

Nhap phan tu thu 5 : 5

Mang khong co so am

-- program is finished running (dropped off bottom) --
```

```
Nhap so phan tu cua mang: 5

Nhap phan tu thu 1: -333

Nhap phan tu thu 2: -4

Nhap phan tu thu 3: -99

Nhap phan tu thu 4: -3

Nhap phan tu thu 5: 2

So am lon nhat la -3 o vi tri thu 4

-- program is finished running (dropped off bottom) --
```

Phân tích cách thực hiện:

Cho max ban đầu là -999999 sau đó nếu âm thì so sánh với số đó, còn không thì thôi.

C.1:

Code:

.data

msg: .asciiz "Nhap xau: "

S: .space 100

\$v0, 54

.text

main:

li

```
la
        $a0, msg
                        # In ra msg
        $a1, S
                        # Luu dia chi xau vao a1
la
        $a2, 100
                        # Max-length = 100
la
syscall
la
        $t4, S
                        # Dia chi dau cua tu ngan nhat
        $t5, S
                        # Dia chi cuoi cua tu ngan nhat
la
                        # Do dai cua tu ngan nhat
la
        $t6, 100
        $t0, S
                        # Dia chi dau cua tu dang xet
la
        $t1, S
la
                        # Dia chi cuoi cua tu dang xet
                        # Do dai cua tu dang xet
la
        $t3, 1
```

Nhap xau

f1: # !!! f1 thuc hien tim vi tri cua tu be nhat

lb \$s1, 0(\$t1) # load ki tu

```
beq \$s1, \$zero, \$s1 # Neu ki tu == \$s1 thi den \$s1 roi thoat
bne $s1, 32, f11# Neu ki tu != space thi den f11
blt $t6, $t3, f12 # If t3(lenght now string) < t6(Min lenght string)
addi $t4, $t0, 0 # Luu dia chi dau cua tu be nhat
sub $t5, $t1, 1
                        # Luu dia chi cuoi cua tu be nhat
addi $t6, $t3, 0 # Luu do dai cua tu be nhat
f12:
addi $t0, $t1, 1 # Dia chi dau cua tu dang xet t0 = t1 + 1
la $t3, 0
                # Do dai tu dang xet t3 = 0
f11:
addi $t1, $t1, 1 # t1 += 1
addi $t3, $t3, 1 # t3 += 1
j f1
                        # jump to f1
f13:
blt $t6, $t3, fout
                                # If t3(lenght now string) < t6(Min lenght string)
addi $t4, $t0, 0 # Luu dia chi dau cua tu be nhat
sub $t5, $t1, 1
                        # Luu dia chi cuoi cua tu be nhat
addi $t6, $t3, 0 # Luu do dai cua tu be nhat
fout:
f2:
        #!!! f2 thuc hien in tu do
li
        $v0, 11
lb
        $a0, 0($t4) # in ra 1 ky tu
syscall
blt $t5, $t4, exit_f2 # Neu t5 < t4 vi tri cuoi thi thoat
addi $t4, $t4, 1 # t4 += 1
j f2
exit_f2:
```

```
.data
          .asciiz "Mhap xau: "
msg:
          .space 100
main:
                     $v0, 54
                                         # Nhap xau
                    $aO, msg
$al, S
                                        # In ra msg
# Luu dia ch
          1a
          la
                    $a2, 100
                                         # Max-length = 100
           syscall
                    $t4. S
                                         # Dia chi dau cua tu ngan nhat
# Dia chi cuoi cua tu ngan nhat
                     $t5, S
          1a
                    $t6, 100
                                          # Do dai cua tu ngan nhat
                    $t0, S
$t1, S
                                         # Dia chi dau cua tu dang xet
# Dia chi cuoi cua tu dang xet
          la
                     $t3, 0
                                           # Do dai cua tu dang xet
                    # /// f1 thuc hien tim vi tri cua tu be nhat

1b $s1, 0($t1)  # load ki tu

beq $s1, $zero, $f13  # Neu ki tu == \0 thi den $f13 roi thoat

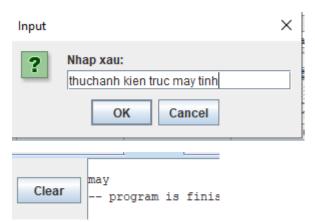
bne $s1, $2, $f11  # Neu ki tu != space thi den $f11

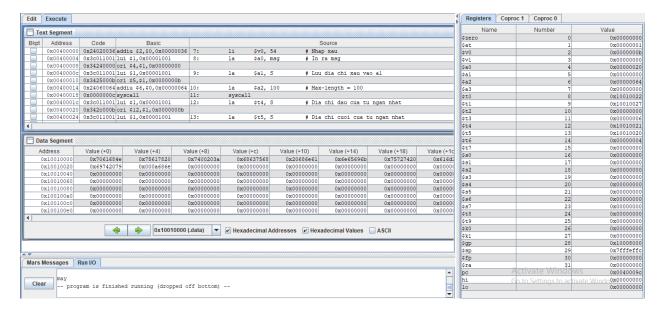
blt $t6, $t3, $f12  # If t3(length now string) < t6;

add $t4 $t0 0  # Lum dig chi dan cue ki
          f1:
                                          , $t3 ,f12 # If t3(length now string) < t6(Min length string) addi $t4, $t0, 0 # Luu dia chi dau cua tu be nhat
                                            sub $t5, $t1, 1
                                                                            # Luu dia chi cuoi cua tu be nhat
# Luu do dai cua tu be nhat
                                            addi $t6, $t3, 0
                                 f12:
                                                              # Dia chi dau cua tu dang xet t0 = t1 + 1
                                 addi $t0, $t1, 1
                                                                 # Do dai tu dang xet t3 = 0
                     f11:
                                                 # t1 += 1
# t3 += 1
# jump to f1
                      addi $t1, $t1, 1
                      addi $t3, $t3, 1
                     j fl
          f13:
                                                                       # If t3(lenght now string) < t6(Min lenght string)
# Luu dia chi dau cua tu be nhat
# Luu dia chi cuoi cua tu be nhat
# Luu do dai cua tu be nhat
                                 blt $t6, $t3 ,fout
                                            addi $t4, $t0, 0
                                             sub $t5, $t1, 1
                                           addi $t6, $t3, 0
                                 fout:
                      # /// f2 thuc hien in tu do
          f2:
                      li $v0, 11
                     1b
                                $a0, 0($t4)
                                                     # in ra 1 ky tu
                      syscall
                      blt $t5, $t4, exit_f2 # New t5 < t4 vi tri cuoi thi thoat
                      addi $t4, $t4, 1
                                                       # t4 += 1
                     j f2
  exit_f2:
```

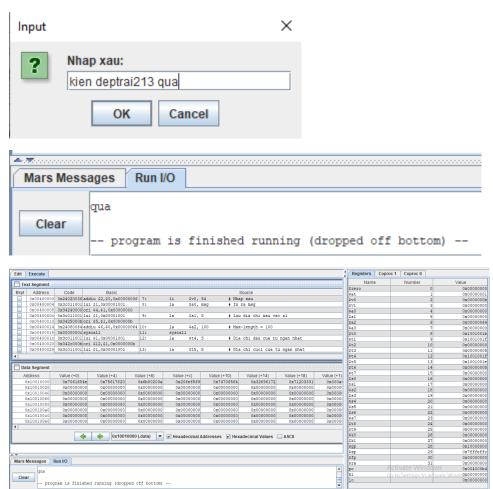
Run:

1:





2:



Phân tích cách thực hiện:

Ví dụ ta có chuỗi: kien deptrai213 qua

Ý tưởng: Đầu tiên ta có 4 con trỏ (Đầu cuối của max và đầu cuối của từ đang xét)

Con trỏ từ hiện tại:

