HO CHI MINH UNIVERSITY OF TECHNOLOGY

FACULTY OF COMPUTER SCIENCE AND ENGINEERING

COMPUTER ARCHITECTURE

Practical session - Week 3

Lecturers: Assoc. Prof. Dr. Cuong Pham Quoc
Mr. Kieu Do Nguyen Binh

Class: CC04 Group: 8

Members: Nguyễn Minh Hùng	Student ID: 1952737
Phạm Nhật Hoàng	Student ID: 1952737
Nguyễn Bá Minh Hưng	Student ID: 1952737

Question 1. Write a MIPS program with the following requirements:
1. Declare an integer array with 10 synthetic data elements.
2. Calculate the sum of all array elements.
3. Print the result to the terminal.

```
.text
.globl main

main:

la      $s0,array     # load array's address in $s0
li      $t0,0         # initialize sum = 0

li      $t1,0         # count = 0
loop:
beq     $t1,10,exit
lw      $t2,0($s0)
add     $t0,$t0,$t2
addi    $s0,$s0,4
addi    $t1,$t1,1
j loop

exit:
li    $v0,4         # print_string
la    $a0, msg
      syscall
li      $v0,1
move    $a0,$t0        # print sum
        syscall





.data
msg: .asciiz "Result: "
array: .word 1,2,3,4,5,6,7,8,9,10
```

mips1.asm

Question 2. Write a MIPS program with the following requirements:
1. Declare an integer array with 10 synthetic data elements.
2. Calculate the sum of all odd elements (a[1]], a[3],...).
3. Calculate the sum of all even elements (a[0], a[2],...).
4. Print the results to the terminal.

```
.text
.globl main

main:

la          $s0,array           # load array's address in $s0
li          $t0,0               # initialize even_sum = 0
li          $t1,0               # initialize odd_sum = 0
li          $t2,0               # count = 0
loop:
beq         $t2,10,exit         # count < 10 ?
lw          $t3,0($s0)

xori        $t4,$t2,1           # count XOR 1 == count + 1 ? -> even
addi        $t5,$t2,1

beq         $t4,$t5,even
add         $t1,$t1,$t3
j next
even:
add         $t0,$t0,$t3
next:
addi        $t2,$t2,1
addi        $s0,$s0,4
j loop

exit:

li     $v0,4                # print_string
la     $a0, msg
       syscall
li          $v0,1
move        $a0,$t0             # print even_sum
            syscall
li     $v0,4                # print space
la     $a0, msr
       syscall
li          $v0,1
move        $a0,$t1             # print odd_sum
            syscall




.data
msg:  .asciiz "Result: Even: "
msr:  .asciiz "\t Odd: "
array:  .word  1,2,3,4,5,6,7,8,9,10
```

mips2.asm

Question 3. Write a MIPS program that receives 10 integer numbers from users through the terminal and store these numbers into an array. Print the sum of all array elements to the terminal.

```
.text
.globl main

main:

li              $v0,4
la              $a0,msg     #Print string
                syscall

la              $s0,arr
li              $t1,0       #counter for loop $t1: i

loop:
beq             $t1,10,exit    # i <10

li              $v0,5
                syscall             #input
move            $t0,$v0

sw              $t0,0($s0)     #store
addi            $s0,$s0,4

addi            $t1,$t1,1
j loop

exit:
li              $t1,0
li              $t3,0

addi            $s0,$s0,-4


loop1:   #loop to compute sum
beq             $t1,10,exit1

lw              $t2,0($s0)
add             $t3,$t3,$t2
addi            $s0,$s0,-4

addi            $t1,$t1,1

j loop1

exit1:
li              $v0,4
la              $a0,msr     #Print string
                syscall

li              $v0,1
move            $a0,$t3
                syscall

.data
msg:   .asciiz "Input: \n"
msr:   .asciiz "Sum = "
arr:  .word   0:10
```

mips3.asm

Question 4. Write a MIPS program with the following requirements:
1. Declare an integer array with 10 synthetic data elements.
2. Print a sentence to terminal to request an integer number that is greater than 0 and less than 10 (assume that user strictly follow this rule).
3. Print value of the element at index collected from the previous step.

```
.text
.globl main

main:

li     $v0,4          # print_string
la     $a0, msg
       syscall

li          $v0,5          # read int
            syscall
move        $t1,$v0        # $t1 : index from user

la          $s0,array      # load array's address in $s0


loop:
beq         $t1,0,exit
addi        $s0,$s0,4
addi        $t1,$t1,-1
j loop

exit:
lw          $t2,0($s0)

li     $v0,4          # print_string
la     $a0, msr
       syscall
li          $v0,1
move        $a0,$t2         # print result
            syscall





.data
msg: .asciiz "Enter the index: "
msr: .asciiz "Result: "
array: .word 1,2,3,4,5,6,7,8,9,10
```

mips5.asm

Question 5. Write a MIPS program that reverses an 10 elements integer array. For example, the array initially stores 1, 3, 5, 7, 9, 11, 13, 15, 17, 19 , the program will change the array to be 19, 17, 15, 13, 11, 9, 7, 5, 3, 1.

```
.text
.globl main

main:

li    $t1,0      # counter i=0
la    $s0,arr    # load address arr to $s0 : arr[0]

addi  $s1,$s0,36   # $s1: arr[9]

loop:
beq   $t1,5,exit   # check condition i < 5 ?

# swap 2 elements
lw  $t0,0($s1)
lw  $t2,0($s0)

sw  $t0,0($s0)
sw  $t2,0($s1)

addi  $s0,$s0,4
addi  $s1,$s1,-4

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

j  loop

exit:

la  $s3,arr    # load address of arr[0] to $s3

li    $v0,4
la    $a0,msr     #Print " Output array: "
syscall

li    $t1,0      # counter i=0

loop1:
beq   $t1,10,exit1   # check condition i < 10 ?
lw  $t3,0($s3)       # load elements from arr

li    $v0,1
move  $a0,$t3       # print the elements
syscall

li    $v0,4
la    $a0,mst     #Print space
syscall

addi  $s3,$s3,4
addi  $t1,$t1,1
j  loop1

exit1:
```

```
li    $v0,4
la    $a0,msg     #Print endline
syscall

.data
arr: .word 1,3,5,7,9,11,13,15,17,19
msr: .asciiz "Output array: "
mst: .asciiz "  "
msg: .asciiz "\n"
```

mips4.asm