Computer Architecture Lab – Week 5's report

Full name	Student ID	Works
Trần Quốc Việt	1953097	Done in lab
Lý Kim Phong	1952916	Done in lab
Nguyễn Đình Khương Duy	1952207	Done in lab

Question 1.

a. mflo moves content of lo to register.
   mfhi moves content of hi to register.

b. $s0 = 0x40000000
   $s1 = 0x00000001

c. $s0 = 0x00000001
   $s1 = 0x00000003

d. In multiplication, lo keeps result of least-significant 32-bits and hi keeps result of most-significant 32-bits.

e. In division, lo keeps result of quotient and hi keeps result of remainder.

f. hi and lo is not a part of 32 general purpose register of MIPS processors.

Question 2.

```
.text
.globl main

main:
      # Print out msg1
      li $v0, 4
      la $a0, msg1
      syscall

      #Read input and save to
      li $v0, 5
      syscall
      move $t0, $v0

      #Initialize register
log2: addi $t1, $zero, 0 #Counter for log 2
      addi $t2, $zero, 0 #int ret = 0;
```

```
loop :
      div $t3, $t0, 2
      beq $t3, 0, Exit
      div $t0, $t0, 2
      addi $t2, $t2, 1
      j loop

Exit :
      li $v0, 4
      la $a0, msg2
      syscall

      li $v0, 1
      move $a0, $t2
      syscall

.data
msg1 : .asciiz "Enter Input : "
msg2 : .asciiz "Result is : "
```

Question 3.

```
      .text
      .globl main
main:
      li $t1, 0 #a
      li $t2, 10 #b
      li $t3, 7 #c

      li $v0,4 # print_string syscall code = 4
      la $a0, msg1 # load the address of msgsyscall
      syscall

      li $v0, 5 # read_int syscall code = 5
      syscall
      move $t0, $v0 #t0 = input

      beq $t0, 0, case0 #check if input == 0
      beq $t0, 1, case1 #check if input == 1
      beq $t0, 2, case2 #check if input == 2
      beq $t0, 3, case3 #check if input == 3
      beq $t0, 4, case4 #check if input == 4

      li $v0,4 # print_string syscall code = 4
      la $a0, msg2 # load the address of msgsyscall
      syscall
      j EXIT

case0 :
      add $t1, $t2, $t3
      j END
```

```
case1 :
      sub $t1, $t2, $t3
      j END

case2 :
      mult $t2, $t3

      #print high
      li $v0,4 # print_string syscall code = 4
      la $a0, msg3 # load the address of msgsyscall
      syscall

      li $v0, 1
      mfhi $a0
      syscall

      #print low
      li $v0,4 # print_string syscall code = 4
      la $a0, msg4 # load the address of msgsyscall
      syscall

      li $v0, 1
      mflo $a0
      syscall

      j EXIT

case3:
      div $t2, $t3
      mflo $t1
      j END

case4:
      div $t2, $t3
      mfhi $t1
      j END

END :
      li $v0, 1
      move $a0, $t1
      syscall

EXIT :

      .data
msg1: .asciiz "Enter input : "
msg2: .asciiz "please input an another integer number\n"
msg3: .asciiz "HI word is : "
msg4: .asciiz "\nLO word is : "
```