Scot Matson
CS 47, Section 01
Homework I

Problem #1

1(a) [add r2 r5 r6 0x2] # opcode: 0x2 / funct: 0x2
0010 010 101 110 010 = 0010 0101 0111 0010 = **0x2572**

1(b) [sub r1 r7 r7 0x3] # opcode: 0x2 / funct: 0x3
0010 001 111 111 011 = 0010 0011 1111 1011 = **0x23FB**

1(c) [addi r1 r2 0x3A] # opcode: 0x4
0100 001 010 11 1010 = 0100 0010 1011 1010 = **0x42BA**

1(d) [ori r2 r3 0x1B] # opcode: 0x3
0011 010 011 01 1011 = 0011 0100 1101 1011 = **0x34DB**

1(e) [jmp 0x23C] # opcode: 0x5
1000 0010 0011 1100 = **0x523C**

1(f) [jal 0x00F] # opcode: 0x6
1001 0000 0000 1111 = **0x600F**

Problem #2

2(a) [addi r1 r2 0x3A] # opcode: 0x4
11 1010 = 1111 1111 1111 1010 = **0xFFFA**

2(b) [ori r2 r3 0x1B] # opcode: 0x3
01 1011 = 0000 0000 0001 1011 = **0x001B**

Problem #3

3) I am Frank born on 28Sep2085
therefore I am a computer guru
and my id is Frank_28Sep208
I am 30 year old and love to eat pizza

Problem #4

```
# Macro: print_str
# Usage: print_str(<string address>)
.macro print_str($arg)
        li      $v0, 4
        la      $a0, $arg
        syscall
.end_macro

# Macro: read_int
# Usage: read_int(<reg>)
.macro read_int($arg)
        li      $v0, 5
        syscall
```

```
        move   $arg, $v0
.end_macro

# Macro: print_reg_int
# Usage: print_reg_int(<reg>)
.macro print_reg_int($arg)
        li      $v0, 1
        move   $a0, $arg
        syscall
.end_macro

# Macro: print_hi_lo
# Usage: print_hi_lo(<str addr>, <str addr>, <str addr>, <str addr>)
.macro print_hi_lo($strHi, $strComma, $strLo, $strEqual)
        print_str($strHi)
        print_str($strEqual)
        mfhi    $t1
        print_reg_int($t1)
        print_str($str_Comma)
        print_str($strLo)
        print_str($strEqual)
        mflo    $t1
        print_reg_int($t1)
.end_macro

# Macro: exit
# Macro: exit
.macro exit
        li      $v0, 10
        syscall
.end_macro

.data
        msg1:           .asciiz  “Enter number for Hi ? “
        msg2:           .asciiz  “Enter number for Lo ? “
        beforeSwap:     .asciiz  “Before swapping “
        afterSwap:      .asciiz  “After swapping “
        strHi:          .asciiz  “Hi”
        strLo:          .asciiz  “Lo”
        strComma:       .asciiz  “, “
        strEqual:       .asciiz  “ = “
        newline:        .asciiz  “\n”

.text
.globl main
main:           # Receive integer for Hi
                print_str(msg1)
                read_int($t1)
                mthi    $t1
```

```
        # Receive integer for lo
        print_str(msg2)
        read_int($t1)
        mtlo    $t1

        # Pre-swap Hi/Lo values
        print_str(beforeSwap)
        print_hi_lo(strHi, strComma, strLo, strEqual)

        print_str(newline)

        # Perform swap
        mfhi    $t1
        mflo    $t2
        mthi    $t2
        mtlo    $t1

        # Post-swap Hi/Lo values
        print_str(afterSwap)
        print_hi_lo(strHi, strComma, strLo, strEqual)

        # System exit
        exit
```

Problem #5

```
# Macro: push
# Usage: push(<reg>)
.macro push($arg)
        sw      $arg, 0x0($sp)
        addi    $sp, $sp, -4
.end_macro

# Macro: pop
# Usage: pop(<reg>)
.macro push($arg)
        addi    $sp, $sp, +4
        lw      $arg, 0x0($sp)
.end_macro
```

Problem #6

6(a) **0x615AA5EFAF25**
6(b) **0x25AFEFA55A61**

Problem #7

7(a) **742**
7(b) **ReReTiReFaSoReTi**