

## EE 2310 Homework #6 Solutions – Writing Loop Programs

*Note: All CLO's in this problem set tie to ABET program-level criterion a.*

Write the following programs as directed. Once again, you need to write and run the programs under your SPIM emulator. **MAKE SURE THAT THEY RUN CORRECTLY!** On the homework due date, email the programs as NotePad attachments, directly to your TA. The questions can be listed and answered in the email itself. **NOTE: SOME ANSWERS ARE REQUESTED IN THE PROBLEMS BELOW. INCLUDE THESE ANSWERS IN THE BODY OF YOUR EMAIL TO THE TA WITH THE ATTACHED PROGRAMS.**

1. (CLO 5—Assy Lang.) Compose a loop program to examine the eight bytes in the words y and z. If a byte is the code for a decimal number, output it to the console. Otherwise, simply ignore it and examine the next byte. The program must be a loop. That means that you must write a loop that will execute eight times to examine each byte, output if necessary, and continue. End the program as usual with a syscall 10.

After completing the program answer this question:

- 1.1. What numbers are output?

329. Note the sequence, due to your Little Endian computer!

Note: You do NOT have to load a word and partition off each byte, to be able to test it. There is an easier way. Think about it!

```
.text
main: la $t2,y
      li $t1,0
      li $v0,11

loop: lb $a0,($t2)
      blt $a0,0x30,next
      bgt $a0,0x39,next
      syscall

next: addi $t1,$t1,1
      beq $t1,8,done
      addi $t2,$t2,1
      j loop

done: li $v0,10
      syscall

.data
y:   .word 0x32774b33
z:   .word 0x2a276c39
```

2. (CLO 5—Assy Lang.) Compose a program to examine the string “Hello, world!\n”, and calculate the total decimal numeric value of all the characters in the string (including punctuation marks), less the numeric value of the vowels. The program should load each letter, add that numeric value to the running total to produce a total sum, and then add the value to a “vowels running total” as well. The program will require a loop. You do not need a counter, since the phrase is null terminated. Remember, punctuation (even spaces!) have a numeric value as well.

What is the total character numeric sum? 848.

```
.text
main: la $t0,str

go:   lb $t1,($t0)
      beqz $t1,print

      add $t2,$t2,$t1
      beq $t1,0x61,sto2
      beq $t1,0x65,sto2
      beq $t1,0x69,sto2
      beq $t1,0x6f,sto2
      beq $t1,0x76,sto2
next: addi $t0,$t0,1
      j go

sto2: add $t3,$t3,$t1
      j next

print: sub $a0,$t2,$t3
       li $v0,1
       syscall
       li $v0, 10
       syscall

.data
str:  .asciiz "Hello, world!\n"
```

3. (CLO 5—Assy Lang.) The word declared as “num1” contains four ASCII characters. If you were simply to output them using an lb command using \$a0 and syscall 11, They would display backwards, since your computer has Little Endian memory organization. Construct a program that will do the following: Print out the characters in the order given (i.e. reverse). Then, use the stack to reverse the data sequence, i.e., in correct order. Remember that you only store words on the stack!

You will need three small loops: A first loop to print the character out in given order, a second loop to store the four data characters (bytes) on the stack as words, and a third to pop the words off the stack and print the characters in correct order. Yes, you WILL need a counter in the loops, although only to count to 4. When addressing the stack, use the “points to the last filled location” convention for the stack pointer (\$29). Note: If you pop each word off the stack into \$a0, then execute a syscall 11, you will print out the least significant byte of the word as an ASCII character (which is the character you want!).

What is printed out in correct order? UTD!

```
.data
num1: .word 0x55544421

.text
main: li $v0,11
      la $t1,num1
p1:   lb $a0,($t1)
      syscall
      addi $t9,$t9,1
      beq $t9,4,next
      addi $t1,$t1,1
      j p1
next: li $a0,0x0a
      syscall

      la $t1,num1
p2:   lb $a0,($t1)
      sub $sp,$sp,4
      sw $a0,($sp)
      sub $t9,$t9,1
      beqz $t9,rev
      addi $t1,$t1,1
      j p2
rev:  li $v0,11
l2:   lw $a0,0($sp)
      addi $sp,$sp,4
      syscall
      addi $t9,$t9,1
      beq $t9,4,done
      j l2
done: li $v0,10
      syscall
```

4. (CLO 5—Assy Lang.) Construct a program that compares the five numbers in the list below and outputs only the positive numbers (i.e., >0).

Note that your loop will need a counter to determine when you have analyzed the five words. When the analysis is complete, end the program as usual. Note: This must be a loop program.

Output a carriage return between numbers to make a neat list.

List the numbers that you output. Note that they will be decimal values!

23505870  
2147480716  
808562423

```
.text
main: li $t1,5
      la $t0,v
loop: lw $a0,0($t0)
      blez $a0,next
      li $v0,1
      syscall
      li $v0,11
      li $a0,0x0a
      syscall
next: sub $t1,$t1,1
      beqz $t1,done
      addi $t0,$t0,4
      j loop

done: li $v0,10
      syscall

.data
v:    .word 0x0166abce
w:    .word 0xafc7d12e
x:    .word 0x7fff48c
y:    .word 0x3031aef7
z:    .word 0xfeff0108
```