

**CS 2734, 26 April 2000****Lab Quiz 9****Interrupts**

Consider the following MIPS assembler program, including a very small exception handler. Execution starts at line 30, which then calls main. (The `-notrap` option must be used.)

```
1 # Program for Quiz 9: Exception Handling
2 main:  la      $t1, Data
3        lw      $t4, 0($t1)
4        li      $v0, 1          # syscall 4 (print_int)
5        add     $a0, $0, $t4
6        syscall
7        add     $t5, $0, $0
8        lw      $t6, 0($t5)
9        li      $v0, 4          # syscall 4 (print_str)
10       la      $a0, All        # print "That's all!"
11       syscall
12       jr      $ra            # return from main
13       .data
14 Data:  .word   314159265
15 All:   .asciiz "\nTh-th-th-that's all folkes!\n"
16 ##### Start of Trap Handler #####
17       .kdata
18 Duhh:  .asciiz "\nDuhh-hhhhh!\n"
19       .ktext  0x80000080
20       li      $v0 4          # syscall 4 (print_str)
21       la      $a0 Duhh      # print "Duhh-hhhhh!"
22       syscall
23       mfc0   $k0 $14
24       rfe
25       addiu  $k0 $k0 4
26       jr     $k0
27 # Standard startup code.  Invoke the routine main with no arguments.
28       .text
29       .globl __start
30 __start: jal main          # start up main
31       li      $v0 10
32       syscall            # syscall 10 (exit)
```

1. Starting with line 2, describe what is printed when this code segment executes lines 2 through 12. Will `Duhh-hhhhh!` be printed and if so, why?

2. What (if anything) is line 26 (`jr $k0`) doing? What would happen if line 25 (`addiu $k0 $k0 4`) were omitted?