

CPSC 359
ARM & RPi – Practice Questions

1. For an Empty, Ascending (EA) stack, write appropriate operations for pop, push, and top (top simply returns the top value on the stack without changing the stack.)
2. Write a top operation for the remaining three types of stacks (FD, FA, and ED)
3. Repeat question 2 for the other operations
4. Assume the following declaration in the data section:
array .word 0, 1, 0, 1, 0, 1, 0, 1
What is the value in r0 after the following code executes?

```
LDR r0, =array
MOV r1, #2
ADD r1, r1, LSL #2
LDRB r2, [r0, r1]!
```

5. In 4, what is the value in r1 after the code executes?
6. In 4, what is the value in r2 after the code executes?
7. What is the value in r0 after the following code segment executes?

```
MOV r0, #0
MOV r1, #1
MOV r2, #2
TST r1, r2
MOVEQ r0, #12
B L1
MOV r0, #11
L1:
CMP r1, r2
MOVNE r0, #10
```

8. What is the address stored by the instruction STR r1, [r2], -r3, LSL #2? After the instruction executes, what is the value of r2?
9. After the instruction STR r1, [r2], -r3, LSL #2 executes, what is the value of r2?
10. Which bit of r0 is changed after the following instructions execute?
MOV r1, #2
MOV r2, r1, LSL #1
EOR r0, r2
11. Write a single instruction that does not use multiplication instructions in order to multiply r0 by 9.

12. Consider the STM32F103 {r5-r7, r0, r1} instruction. At which address (in terms of sp) is each of the registers stored, assuming an ED stack?
13. What meaningful name would you give the following subroutine, assuming r2 is a positive parameter?

```
subQuiz:
    LDMFD sp!,{r1,r2}
    MOV r1, r1, LSL r2
    MOV pc, lr
```
14. Write a code segment of no more than three instructions that enables SPI1.
15. Write a SNES signal that indicates that every other button on the SNES controller is pressed.
16. Write a code segment that reads from pin #3.
17. Write an appropriate return statement from a data abort exception handler.
18. Write a code segment that enables interrupts for IRQ.
19. Assuming you have a subroutine paintPixel, which sets a pixel at a given physical offset passed in r0. Write a routine that draws a right-angle isosceles, whose base is a horizontal line. Use any appropriate dimensions.
20. Assume the declarations

```
.equ AUX_MU_LSR_REG,      0x20215054
.equ AUX_MU_IO_REG,       0x20215040
```

Answer the following questions based on the following code:

```
ldr      r2, =AUX_MU_LSR_REG
loop:
    ldr      r1, [r2]
    tst      r1, #0x1
    beq      loop

    ldr      r2, =AUX_MU_IO_REG
    ldr      r0, [r2]
```

 - a. When does the loop break?
 - b. When bit 0 in LSR is set, what does this mean?
 - c. What do the two lines after the loop do?
 - d. When bit 5 in LSR is set, what does this mean?
21. Write a code segment that sets the function of pin# 3 to input.
22. Write the subroutine setGPIOFunc(pin#, functionCode) in ARM assembly (it sets the function for a generic pin).