Quiz #5
Wednesday, May 17th

1. [5 pts] The code below shifts a 64-bit integer right by 1 bit. Show how the same general approach can be used to shift left by 1 bit.

LDRD R0,R1,x64	LDRD R0,R1,x64
LSR R0,R0,1	LSL R1,R1,1
ORR R0,R0,R1,LSL 31	ORR R1,R1,R0,LSR 31
LSR R1,R1,1	LSL R0,R0,1
STRD R0,R1,x64	STRD R0,R1,x64

2. [5 pts] The code below shifts a 64-bit integer right by 1 bit. Show how the same general approach can be used to shift right by 5 bits.

LDRD R0,R1,x64	LDRD R0,R1,x64
LSR R0,R0,1	LSR R0,R0,5
ORR R0,R0,R1,LSL 31	ORR R0,R0,R1,LSL 27
LSR R1,R1,1	LSR R1,R1,5
STRD R0,R1,x64	STRD R0,R1,x64

3. [5 pts ea] Give a short sequence of ARM assembly language instructions to:

Set bit 5 of 32-bit variable x to 1 without affecting any other bits.	LDR R0,x ORR R0,R0,1<<5 STR R0,x
Clear bit 5 of 32-bit variable x to 0 without affecting any other bits.	LDR R0,x BIC R0,R0,1<<5 STR R0,x
Change the value of bit 5 of 32-bit variable x without affecting any other bits.	LDR R0,x EOR R0,R0,1<<5 STR R0,x

1. [5 pts] Register R0 contains packed data. There is a 10-bit signed 2's complement integer in bits 9-18. Which of the following instructions puts that integer into register R1?

`UBFX R1,R0,9,10`	`SBFX R1,R0,9,10`	`ASR R1,R0,9`

4. [5 pts] Which of the following sequences creates a 32-bit word in R0 that has 1's in bits 9-18 and 0's everywhere else?

`LDR R2,=1<<18`
`LDR R3,=1<<9`
`SUB R0,R2,R3`
	`LDR R2,=1<<19`
	`LDR R3,=1<<9`
	`SUB R0,R2,R3`
		`LDR R1,=-1`
		`LSL R1,R1,10`
		`ROR R0,R1,18`

5. [5 pts] Write a function in ARM assembly language that returns the negative of its only parameter. For example, if the parameter is +5, it should return -5; if the parameter is -5, it should return +5. Do not use a SUB or NEG instruction. The function prototype is

```
int32_t Negate(int32_t) ;
```

```
Negate:    MVN   R0,R0
           ADD   R0,R0,1
           BX    LR
```