1. You can multiply register R0 by the binary constant 01011110 using 5 shifts and 4 additions. However, you can reduce the total number of operations if you also use subtractions. Give a minimal length sequence of ARM Cortex-M4 instructions to do this. *(Note: This can be done in 3 instructions.)*

```
LSL   R1,R0,6          // R1 ← 64*R0
ADD   R1,R1,R0,LSL 5   // R1 ← 64*R0 + 32*R0
SUB   R0,R1,R0,LSL 1   // R0 ← 64*R0 + 32*R0 – 2*R0
```

2. Give a minimal length sequence of ARM Cortex-M4 instructions to multiply the contents of register R0 by each of the following values without using a multiplication instruction.

(a) 3
```
ADD   R0,R0,R0,LSL 1   // R0 ← R0 + 2*R0
```

(b) 5
```
ADD   R0,R0,R0,LSL 2   // R0 ← R0 + 4*R0
```

(c) 7
```
RSB   R0,R0,R0,LSL 3   // R0 ← 8*R0 – R0
```

(d) 9
```
ADD   R0,R0,R0,LSL 3   // R0 ← R0 + 8*R0
```

(e) 11
```
LSL   R1,R0,3          // R1 ← 8*R0
ADD   R1,R1,R0,LSL 1   // R1 ← 8*R0 + 2*R0
ADD   R0,R1,R0         // R0 ← 8*R0 + 2*R0 + R0
```

Here's an even better solution I found:

```
ADD   R1,R0,R0,LSL 2   // R1 ← R0 + 4*R0
ADD   R0,R0,R1,LSL 1   // R0 ← R0 + 2*R1
```

(f) 13
```
LSL   R1,R0,3          // R1 ← 8*R0
ADD   R1,R1,R0,LSL 2   // R1 ← 8*R0 + 4*R0
ADD   R0,R1,R0         // R0 ← 8*R0 + 4*R0 + R0
```

A better solution found by a student!!

```
ADD   R1,R0,R0,LSL 4   // R1 ← 17*R0
SUB   R0,R1,R0,LSL 2   // R0 ← 17*R0 – 4*R0
```

3. Suppose you need to divide an unsigned 8-bit integer variable X by 9, but there is no divide instruction. If you use reciprocal multiplication, what constant should you multiply times X?

$2^8/9 = 256/9 = 28.444444$ ➔ use 28

4. What integer quotient does reciprocal multiplication produce when trying to divide the 8-bit integer X by 3 when X has the value 75?

   $75 * (2^8/3) \approx 75 * 85 = 6375_{10} = 18E7_{16}$
   $Quotient = 18_{16} = 24_{10}$

5. Suppose you need to divide an unsigned N-bit integer X by a constant K, but there is no unsigned integer divide instruction. If you use reciprocal multiplication…

   (a) Give an expression for the value of the constant multiplier. $2^N/K$

   (b) How many bits will be in the product? 2N

   (c) Which bits of the product hold the integer quotient? Most-significant half

6. Suppose you have an 8-bit processor with 8-bit registers and a single multiply instruction that produces an unsigned 16-bit product of two 8-bit operands. There is no divide instruction, so you use reciprocal multiplication when you need to divide an integer variable by a constant.

   (a) If the constant divisor is 5, what integer multiplier do you use?

   $2^8/5 = 256/5 = 51.2 \rightarrow$ use 51

   (b) If the variable contains the value 50, what will be the integer quotient?

   $50*51 = 2550_{10} = 09F6_{16}$
   $Quotient = 09_{16} = 9_{10}$