Xavier Kuehn

1. Multiply R0 by 01011110 (94) without using the MUL instruction.

```
// assume that R1 holds a value of zero initially
ADD  R1,R1,R0,LSL 6      // R1 <- 0 + 64R0 = 64R0
ADD  R1,R1,R0,LSL 5      // R1 <- 64R0 + 32R0 = 96R0
SUB  R1,R1,R0,LSL 1      // R1 <- 96R0 - 2R0 = 94R0
```

2. Multiply R0 by 22, 23, and 25 without using the MUL instruction.

```
// assume that R1 holds a value of zero initially
ADD  R1,R1,R0,LSL 4      // R1 <- 0 + 16R0 = 16R0
ADD  R1,R1,R0,LSL 3      // R1 <- 16R0 + 8R0 = 24R0
SUB  R1,R1,R0,LSL 1      // R1 <- 24R0 - 2R0 = 22R0

ADD  R1,R0,R0,LSL 3      // R1 <- R0 + 8R0 = 9R0
ADD  R1,R1,R1,LSL 2      // R1 <- 9R0 + 18R0 = 27R0
SUB  R1,R1,R0,LSL 2      // R1 <- 27R0 - 4R0 = 23R0

ADD  R1,R0,R0,LSL 5      // R1 <- R0 + 32R0 = 33R0
SUB  R1,R1,R0,LSL 3      // R1 <- 33R0 - 8R0 = 25R0
```

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

Answer: **25**

6. Without using the divide instruction, write an assembly function to divide a signed integer by 2^k.

```
int32_t Remainder(int32_t s32, unt32_t k);
// can use reciprocal multiplication instead of division
Remainder:LDR  R2,=0xFFFFFFFF16    // load R2 with 2^32
          LSR  R3,R2,R1            // 2^32 >> k (2^32 / 2^k)
          MUL  R3,R0,R3            // s32 • (2^32 / 2^k)
          LSR  R3,R3,32            // s32 • (2^32 / 2^k) >> 32
          LDR  R2,=(1 << R1)       // R2 <- 2^k
          MLS  R0,R2,R3,R0         // R0 <- R0 - R4*R3
          BX   LR
```

8. How many clock cycles does using reciprocal multiplication save?

unsigned division by 14: **~6 clock cycles**

signed division by +14: **~7 clock cycles**

signed division by -14: **~7 clock cycles**

9. Write an assembly program that returns the quotient of an unsigned 32-bit integer and 14.

```
uint32_t DivBy14(uint32_t dividend);
DivBy14:  LDR   R1,=613566757
          UMULL R2,R1,R1,R0
          SUBS  R0,R0,R1
          ADD   R0,R1,R0,LSR 1
          LSRS  R0,R0,3
          BX    LR
```