CLASSICAL AND QUANTUM COMPUTING
EXERCISE III

1. There are two ways to perform binary division, either by repeated subtraction or using a shift-and-subtract principle. The latter is normally used in practice as it is much faster.

   Division by repeated subtraction is performed by subtracting the divisor from the dividend until the result of the subtraction is negative. The resultant quotient is given by the number of subtractions required. The remainder is obtained by adding the divisor to the negative result.

   The shift-and-subtract method of division is performed by succesively subtracting the divisor from the appropriate shifted dividend and inspecting the sign of the remainder after each subtraction. If the sign of the remainder is positive then the value of the quotient is 1, but if the sign of the remainder is negative then the value is 0 and the dividend is restored to its previous value by adding the divisor. The divisor is then shifted one place to the right, and the next significant bit of the dividend is included and the operation repeated until all bits in the dividend have been used. To simplify the method further, instead of adding the divisor when the subtraction yields a negative result, we can add the divisor shifted right by one position.

   For example, consider the division of 90 by 9 viewed as 8 bit numbers. 90 is given as

   ```
   01011010
   ```

   in binary and 9 is given as

   ```
   00001001
   ```

   in binary.

Then

```
        0|1011010
-00001001
 111101111|011010          -> negative -> 0
+ 00001001
  111110000|11010          -> negative -> 0
+  00001001
   111110011|1010          -> negative -> 0
+   00001001
    111111001|010          -> negative -> 0
+    00001001
     000000100|10          -> positive -> 1
-     00001001
      111110111|0          -> negative -> 0
+      00001001
       000000000|          -> positive -> 1
-       00001001
        11110111           -> negative -> 0
+       00001001
        00000000           Remainder
```

The least significant bit is given last. Thus the answer is `00001010`. Write a C++ program which implements this algorithm.