CS302 Loops

The goal of this exercise is to work with conditions as well as branching and looping statements.

1.) Prime Numbers

Write a class method named `isPrime` that is passed an integer and returns true if that integer is prime.

```
public static boolean isPrime( int n ) {
   int divisor = 1;
   while ( divisor <= n ) {
      if ( ( n % divisor == 0 ) && ( divisor != 1 ) && (divisor != n) )
         return false;
      divisor = divisor + 1;
   }
   return true;
}

public static boolean isPrime( int n ) {
   int divisor = 2;
   while ( divisor < n/2 ) {
      if (  n % divisor == 0 )
         return false;
      divisor++;
   }
   return true;
}

public static boolean isPrime( int n ) {
   int divisor = 2;
   int sqroot = (int) Math.sqrt(n);
   while ( divisor <= sqroot ) {
      if (  n % divisor == 0 )
         return false;
      divisor++;
   }
   return true;
}
```

2.) Break the Code

Consider the following class:

```
class Lock {

   private long combination;

   // Constructor randomly determines a combination (long integer)
   public Lock( ) { ... }

   // returns the number of digits in the combination
   public int getLength() { ... }
```

```
    // digit:    a single digit, 0 to 9
    // position: the digit's position in the combination, 1 to length-of-combination
    // returns true if the digit is correct for position specified
    public boolean checkDigit( int digit, int position ) { ... }
}
```

Write a code fragment that determines and displays the combination of `lock`.

```
InputBox   in = ... // assume it is properly constructed
OutputBox out = ... // assume it is properly constructed
Lock     lock = new Lock(in.getLong("enter the length of the lock"));

int n = lock.getLength();

for ( int position = 1; position <= n; position++ ) {
   boolean found = false;
   for ( int digit = 0; digit < 10 && ! found; digit++ ) {
      if ( lock.checkDigit( digit, position ) ) {
         // we have found the correct digit for this position
         out.print(digit);
         found = true;
      }
   }
}
out.printLine();
```