Introduction to Programming
Lab Exercise for Week 18 (Trimester 2, week 5)

An exercise on birthdays that uses arrays

This is question 7 from chapter 3 of the textbook.

Study the example in Subsection 3.8.3 that tried to answer the question, How many random people do you have to select before you find a duplicate birthday? The source code for that program can be found in the file BirthdayProblemDemo.java, which is in the T2-Week05 folder on Moodle, and is also given below.

Here is the program:

```
/**
 *  How many random people do you have to select before you find
 *  a duplicate birthday (that is, two peole who were born on the
 *  same day of the same month, but not necessarily in the same
 *  year).  This program simulates the process.  (It ignores the
 *  possibility of people born on leap day.)
 */

public class BirthdayProblemDemo {

  public static void main(String[] args) {

       TextIO.putln("This program simulates taking people at random");
       TextIO.putln("until two have been found who were born on the");
       TextIO.putln("same day of the year.\n");

       do {
          birthdayProblem();
          TextIO.put("\nAgain? (Y/N):  ");
       } while ( TextIO.getlnBoolean() );

       TextIO.putln("\n\nOK.  Goodbye.");

   } // end main()


   /**
    * Simulate choosing people at random and checking the day of the
    * year they were born on.  If the birthday is the same as one that
    * was seen previously, stop, and output the number of people who
    * were checked.
    */
   private static void birthdayProblem() {

      boolean[] used;  // For recording the possible birthdays
                       //   that have been seen so far.  A value
                       //   of true in used[i] means that a person
                       //   whose birthday is the i-th day of the
                       //   year has been found.

      int count;       // The number of people who have been checked.
```

```
      used = new boolean[365];  // Initially, all entries are false.

      count = 0;

      while (true) {
            // Select a birthday at random, from 0 to 364.
            // If the birthday has already been used, quit.
            // Otherwise, record the birthday as used.
         int birthday;  // The selected birthday.
         birthday = (int)(Math.random()*365);
         count++;
         if ( used[birthday] )  // This day was found before; It's a
                                // duplicate.
            break;
         used[birthday] = true;
      }

      TextIO.putln("A duplicate birthday was found after "
                                              + count + " tries.");

   } // end birthdayProblem()


} // end class BirthdayProblemDemo
```

Here are some related questions:

- How many random people do you have to select before you find three people who share the same birthday? (That is, all three people were born on the same day in the same month, but not necessarily in the same year.)

- Suppose you choose 365 people at random. How many different birthdays will they have? (The number could theoretically be anywhere from 1 to 365).

- How many different people do you have to check before you've found at least one person with a birthday on each of the 365 days of the year?

Write three programs to answer these questions. Each of your programs should simulate choosing people at random and checking their birthdays. (In each case, ignore the possibility of leap years.)