Leap Year

In the lab1 folder, you should see a file called **LeapYear.java**. This program is supposed to test whether or not a given year is a Leap Year. The user will give a year as a command line parameter (examples given below), and then print out whether or not that year is a leap year, e.g.

```
$ java LeapYear 2000
2000 is a leap year.
$ java LeapYear 1999
1999 is not a leap year.
$ java LeapYear 2004
2004 is a leap year.
$ java LeapYear 2100
2100 is not a leap year.
```

A leap year is either:

- divisible by 400 or
- divisible by 4 and not by 100.

For example, 2000 and 2004 are leap years. 1900, 2003, and 2100 are not leap years.

Your code must declare a method as follows: **public static boolean isLeapYear(int year)**. Make sure to provide a description of the method as a comment.
Your description should be contained by **/**** and ***/**. Comments contained by **/**** and ***/** are also called "Javadoc comments" or just "Javadocs". These comments can span multiple lines if they need the extra space, e.g. the checkLeapYear Javadocs.

Javadocs may contain optional tags, e.g. **@param**. We do not require you to use any tags like this in 61B except the **@source** tag. Use the **@source** tag any time you receive significant help on a project. The **@source** tag is not required for HW or lab, though we recommend it anyway, since it's a good scholarly and professional habit to cite your sources.

Some Java tips:

- The **%** operator implements remainder. Thus, the value of **year % 4** will be 0, 1, 2, or 3.
- The **!=** operator compares two values for inequality. The code fragment **if (year % 4 != 0)** reads as “if the remainder when dividing year by 4 is not equal to 0.”
- When one of the arguments of the **+** operator is a String, the arguments are concatenated as Strings. For example, **"horse" + "babies"** would return **"horsebabies"**