

NAME:

login id:

Precept:

COS 126 Midterm 1 Programming Exam Fall 2011

This part of the exam is like a mini-programming assignment. You will create a program, compile it, create test data for it and run it on your laptop. Debug it as needed. This exam is open book, open browser—but only our course website and booksite! Of course, no internal or external communication is permitted (e.g., talking, email, IM, texting, cell phones) during the exam. You may use code from your assignments or code found on the COS126 website. When you are done, submit your program via the course website using the submit link for Precept Exam 1 on the Assignments page.

Grading. Your program will be graded on correctness, clarity (including comments), design, and efficiency. You will lose a substantial number of points if your program does not compile or if it crashes on typical inputs.

Even though you will electronically submit your code, you must turn in this paper so that we have a record that you took the exam and signed the Honor Code. *Print your name, login ID, and precept number on this page* (now), and write out and sign the Honor Code pledge before turning in this paper. Note: It is a violation of the Honor Code to discuss this midterm exam question with anyone until after everyone in the class has taken the exam. You have 50 minutes to complete the test.

“I pledge my honor that I have not violated the Honor Code during this examination.”

Part 1	/20
Part 2	/10
TOTAL	/30

Your task in this exam is to write two programs that address the following version of the well-known *Birthday Problem*.

Suppose that you are given a long list of people's birthdays. If you examine the entries on the list one by one, how many entries do you have to examine before finding M birthdays that match?

Part 1. Write a program `BirthdayPart1` that takes an `int` value M from the command line and reads from standard input a sequence of `int` values that represent birthdays and are guaranteed to be between 0 and 365 inclusive (one for each of the 366 possible dates, including February 29), stopping and printing out the number of values read as soon as some value has been found to occur M times (or a message indicating that no such value was found). For example, if the file `20birthdays.txt` contains the numbers

```
101 123 342 2 3 45 101 31 89 73 123 84 83 65 23 99 123 245 289 302
```

your program must behave as follows:

```
% java BirthdayPart1 2 < 20birthdays.txt
7 birthdays examined to find 2 occurrences of 101

% java BirthdayPart1 3 < 20birthdays.txt
17 birthdays examined to find 3 occurrences of 123

% java BirthdayPart1 4 < 20birthdays.txt
No birthday occurs 4 times
```

To further test your program (as we will!), you may download our test file

<http://introc.cs.princeton.edu/data/1000birthdays.txt>

and run it to find the number of birthdays examined to find a value that occur 5, 6, 7, 8, and 9 times. Your answers for this file must be 273, 528, 589, 670, and 989, respectively.

Submission. Submit the single file `BirthdayPart1.java` via Dropbox at

https://dropbox.cs.princeton.edu/COS126_F2011/Exam1

(This is the submit link for Precept Exam 1 Part 1 on the Assignments page.) Be sure to click the *Check All Submitted Files* button to verify your submission.

Grading. Your program will be graded on correctness and clarity (including comments).

Part 2. Be sure that you have successfully submitted `BirthdayPart1.java` before attempting this part of the exam (which will take longer and counts for half as many points).

Write a program `BirthdayPart2` that takes `int` values `max` and `T` from the command line. For each value of `M` from 2 to `max`, your program should run `T` experiments. Each experiment consists of generating and counting random birthdays (random numbers with each of the 366 values between 0 and 365 equally likely) until some value is found to occur `M` times.

You must organize your program as follows: Define a static method `experiment()` that takes an `int` argument `M`, runs one experiment and returns an `int` value giving the number of birthdays examined to find one that repeats `M` times. Then write a `main()` that calls `experiment()` within a pair of nested `for` loops.

Your output should be a table that gives the average number of values examined, like the following.

```
% java BirthdayPart2 9 1000
2 24
3 88
4 185
5 314
6 459
7 623
8 780
9 974
```

That is, for each `M` between 2 and `max`, print the average number of birthdays examined (over your `T` experiments for that value of `M`, rounded to the nearest integer) to find a value that occurs `M` times. Note: Your values might be slightly different than the ones above because of the randomness (but they would be likely to match if we were to use a larger value of `T`).

Submission. Submit the single file `BirthdayPart2.java` via Dropbox at

https://dropbox.cs.princeton.edu/COS126_F2011/Exam1

(This is the submit link for Precept Exam 1 Part 2 on the Assignments page.) Again, be sure to click the *Check All Submitted Files* button to verify your submission.

Grading. Your program will be graded on correctness and clarity (including comments) and on your success in structuring the program as specified.