

### LICENSE PLATE NUMBER SEQUENCING

A standard Virginia license plate number consists of three (uppercase) letters and a 4-digit number. Write a C++ program that determines the next license plate number given a preceding license plate number. For example, if the program is given the input “ABC 1234”, it would output “ABC 1235”. Most of the time, this is just a matter of adding one to the numeric portion of the plate number. However, if the input is something like “BAD 9999”, the output would be “BAE 0000”, which is the next plate number in the sequence if all plate numbers were listed in alphabetical order. It becomes slightly more complicated when the numeric portion of the plate number is 9999 **and** it contains the letter ‘Z’. For instance, if the input is “JAZ 9999”, the output would be “JBA 0000”. The plate number “ZZZ 9999” is a special case in that no plate number succeeds it. For this case, just output the message “That is the last plate”.

The input file will be named “Plates.txt” and the first line in that file will be the phrase “Plate Numbers”, which your program should ignore. The file will continue with a list of plate numbers, one plate number per line (three uppercase letters, followed by one or more spaces, followed by a non-negative integer less than 10000). The output file should be named “NextPlates.txt” and should contain the input plate numbers together with the succeeding plate number. At the end of the output file, indicate how many plates were processed by the program. Refer to the next section for header information that should be in the output file, as well as format details (in particular, the numeric portion of a plate number should be listed with leading zeroes, if necessary).

Finally, in case the input file does not exist, output a “File not found” message in the output file.

#### Sample input and corresponding output:

```
Plate Numbers
ABC 1234
XYZ 9999
XYZ 9980
BZZ 9999
ZZZ 9999
PQR 9999
AAA 0000
```

```
Programmer: <put your name here>
CS 1044 Summer I 2004 Project 2

ABC 1234 ---> ABC 1235
XYZ 9999 ---> XZA 0000
XYZ 9980 ---> XYZ 9981
BZZ 9999 ---> CAA 0000
ZZZ 9999 ---> That is the last plate
PQR 9999 ---> PQS 0000
AAA 0000 ---> AAA 0001

Number of plates processed: 7
```

#### Submitting your program:

This program is an exercise in **if** statements and file I/O. It should follow the usual documentation and submission standards: your program must include a header comment, pledge statement, and helpful inline comments explaining key portions of your code. Refer to “Elements of Programming Style” available in the course website for some guidelines. You may also want to refer to the first programming assignment as that contained an example of a program with sufficient documentation. As usual, submissions will be handled and auto-graded through the curator system, but will be hand-graded for compliance and documentation by the GTA.