Exam 2 Practice Assignment

Overview

This is a set of problems to use to prepare for Exam 2. **It is not a "practice exam".** The actual exam will not be this length or have this many questions.
You will NOT hand in any files. This assignment is only practice for Exam 2.

For each problem, you should:

- determine the inputs, outputs, and error conditions
- construct an algorithm
- code your algorithm in Java
- construct a set of test cases and use them to test **and fix** your program

Problems

Problem 1

Your program should take a string that has been encoded using a "secret" code, and decode it. The code works as follows:

- Each letter is decoded using the letter immediately preceding it in the alphabet. "b" becomes "a", "c" becomes "b", etc.
- The letter "a" becomes "z".
- Each digit is decoded using the preceding digit. "1" becomes "0", "2" becomes "1", etc.
- The digit "0" becomes "9".
- Characters that are neither letters nor digits are unchanged.

Regardless of whether the input string contains upper- or lowercase characters, the output (one string) should be in all lowercase.

Example:

```
java Cipher

NFFU NF BU 23 JO UIF CFMM UPXFS

RESULT: "meet me at 12 in the bell tower"
```

Problem 2

Ask the user for the names of three books (three strings). Alphabetize these titles, ignoring the word "the" if it appears at the beginning of a title. Print the **full names** of the books (three strings, including an inital "the" if present) in alphabetical order. Do not make any assumptions about the case (uppercase/lowercase) of the input. However, it does not matter whether your output is in uppercase or lowercase.

Example:

```
java Titles

Hamlet
The Great Gatsby
Great Expectations


RESULT: "Great Expectations"
RESULT: "The Great Gatsby"
RESULT: "Hamlet"
```

Problem 3

Ask the user for a sentence. Output a sentence containing the same words, but in reverse order. You may assume that the input sentence contains no punctuation except for a period at the end. The output sentence should be properly capitalized and ended with a period.

```
java ReverseWords

The quick brown fox jumps over the lazy dog.

RESULT: "Dog lazy the over jumps fox brown quick the."
```

Problem 4

Take the programs you wrote for Problems 1, 2, and 3, as well as the programs you wrote for Assignments 2, 3, and 4. Convert these programs into *methods* that are intended to be invoked by another program. Write these methods in a new module.
Test your methods by writing a simple driver program that invokes each method and prints its return value.

Problem 5

Write a program that asks the user for a list of people's names and birthdays. Birthdays should be input as two numbers, month and day. Use arrays to store this information. Create a Class that represents a person. Display a menu that offers the following choices:
- Given a person's name, report their birthday.
- Given a month, report all people whose birthdays fall in that month.
- Report all sets of people who happen to have the same birthday.

Problem 6

Write a program that keeps track of P.O. boxes at the post office. Ask the user how many P.O. boxes exist, then create an array of that size. For each P.O. box, you should keep track of whether it has been reserved, and the name of the person who reserved it (if any). Create a class that represents a POBox. Display a menu that offers the following choices:
- Determine whether a specific box has been reserved. If so, report the name of the person who reserved it.
- Reserve a box for a new customer. Find a free box and assign it to this person.
- Release a box. This will be done when a customer no longer wants his or her box. The box numbers of all other customers must stay the same.
- Expand the set of boxes. Ask the user for a new, larger number of boxes and create an array of that new size. Transfer all information from the old array to the new one, taking care to keep each person's box

number the same.