**Java 2 - Mid-term project**

**Part 1**

# Lottery Statistics

To play the PowerBall lottery, you buy a ticket that has five numbers in the range of 1–69, and a “PowerBall” number in the range of 1–35. (You can pick the numbers yourself, or you can let the ticket machine randomly pick them for you.) Then, on a specified date, a winning set of numbers are randomly selected by a machine. If your first five numbers match the first five winning numbers in any order, and your PowerBall number matches the winning PowerBall number, then you win the jackpot, which is a very large amount of money. If your numbers match only some of the winning numbers, you win a lesser amount, depending on how many of the winning numbers you have matched.

In the student sample programs for this book, you will find a file named pbnumbers.txt, containing the winning PowerBall numbers that were selected between February 3, 2010 and May 11, 2016 (the file contains 654 sets of winning numbers). Here is an example of the first few lines of the file’s contents:

17 22 36 37 52 24

14 22 52 54 59 04

05 08 29 37 38 34

10 14 30 40 51 01

07 08 19 26 36 15

and so on . . .

Each line in the file contains the set of six numbers that were selected on a given date. The numbers are separated by a space, and the last number in each line is the PowerBall number for that day. For example, the first line in the file shows the numbers for February 3, 2010, which are 17, 22, 36, 37, 52, and the PowerBall number 24. Write one or more programs that work with this file to perform the following: Display the 10 most common numbers, ordered by frequency Display the 10 least common numbers, ordered by frequency Display the 10 most overdue numbers (numbers that haven’t been drawn in a long time), ordered from most overdue to least overdue Display the frequency of each number 1–69, and the frequency of each Powerball number 1–35

**Part 2**

# Payroll Class & Exceptions

Design a Payroll class that has fields for an employee’s name, ID number, hourly pay rate, and number of hours worked. Write the appropriate accessor and mutator methods and a constructor that accepts the employee’s name and ID number as arguments. The class should also have a method that returns the employee’s gross pay, which is calculated as the number of hours worked multiplied by the hourly pay rate. Write a program that demonstrates the class by creating a Payroll object, then asking the user to enter the data for an employee. The program should display the amount of gross pay earned.

Payroll Class Exceptions Programming Challenge 5 of Chapter 6 required you to write a Payroll class that calculates an employee’s payroll. Write exception classes for the following error conditions: An empty string is given for the employee’s name. An invalid value is given for the employee’s ID number. If you implemented this field as a string, then an empty string would be invalid. If you implemented this field as a numeric variable, then a negative number or zero would be invalid. An invalid number is given for the number of hours worked. This would be a negative number or a number greater than 84. An invalid number is given for the hourly pay rate. This would be a negative number or a number greater than 25. Modify the Payroll class so that it throws the appropriate exception when any of these errors occurs. Demonstrate the exception classes in a program.

Part 3

# Ship, CruiseShip, and CargoShip Classes

Design a Ship class that the following members:

* A field for the name of the ship (a string).
* A field for the year that the ship was built (a string).
* A constructor and appropriate accessors and mutators.
* A toString method that displays the ship’s name and the year it was built.

Design a CruiseShip class that extends the Ship class. The CruiseShip class should have the following members:

* A field for the maximum number of passengers (an int).
* A constructor and appropriate accessors and mutators.
* A toString method that overrides the toString method in the base class. The
* CruiseShip class’s toString method should display only the ship’s name and the maximum number of passengers.

Design a CargoShip class that extends the Ship class. The CargoShip class should have the following members:

* A field for the cargo capacity in tonnage (an int).
* A constructor and appropriate accessors and mutators.
* A toString method that overrides the toString method in the base class. The CargoShip class’s toString method should display only the ship’s name and the ship’s cargo capacity.

Demonstrate the classes in a program that has a Ship array. Assign various Ship, CruiseShip, and CargoShip objects to the array elements. The program should then step through the array, calling each object’s toString method.