# Your Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Student Number: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Lab 10a: to be done in class with a partner – due at the end of this lesson

# ProvinceTerritory

Create a class called ProvinceTerritory. It contains two instance variables: a String name and an int population. Create accessors and mutators too. The constructor takes the name and population as parameters and sets them as long as the name isn’t null and the population isn’t negative.

# Canada

Create a class called Canada. It contains an ArrayList of ProvinceTerritory references:



Populate the ArrayList inside the Canada constructor. The following code will be useful:

**provinces.add(new ProvinceTerritory("british columbia", 4400057));**

**provinces.add(new ProvinceTerritory("alberta", 3645257));**

**etc...**

**NOTE: also, somewhere in the middle of all the provinces add in:**

**provinces.add(null);**

**…and code your methods so that they are careful to deal with this, without crashing.**

Create a method called **public int getTotalPopulation()** which uses a for-each loop to add up all of the individual populations to calculate Canada’s total population.

Create a method called **public String getLowestPopulation()** which uses a for-each loop to determine and return the name of the province/territory that has the lowest population.

Create a method called **public ProvinceTerritory getHighestPopulation()** which uses a for-each loop to determine and return the name of the province/territory that has the highest population.

Create a method called **public int getPopulation(String province)** which returns the population of the province (the parameter); if there is no such province, return a constant called NO\_SUCH\_PROVINCE, which is an int set to -1.

Demonstrate your completed project to your instructor. When your instructor is satisfied, your paper will be signed and you can go home. Lab 8b (below) is due at the next lesson, and there is also a quiz next day.

Checked by: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**NOTE:** keep this paper for your instructor to verify your grades later in the course.

NOTE: EVERY SINGLE STUDENT MUST SUBMIT THIS LAB AND GET HIS OR HER PAPER SIGNED….

# Lab 10b: at home, alone – due next class (in person at the end of NEXT lab)

Continue with the Canada class you made in lab 10a, above.

Create a method called **public boolean isProvinceInCanada(String name)** which returns true if there is a province/territory in Canada with the given name (the parameter); otherwise returns false.

Create a method called   
**public ProvinceTerritory[] getProvincesWhoseNameContains(String substring)** which returns an array of the names of all provinces/territories whose name contains substring (the parameter). Hint: use the String class’s **contains()** method.

**public ArrayList<ProvinceTerritory>   
 getMoreProvincesWhoseNameContains(String substring)**

which returns an ArrayList of the names of all provinces/territories whose name contains substring (the parameter). Hint: use the String class’s **contains()** method.

Create a method called   
**public ArrayList<String> getProvincesWhoseNameStartsWith(char letter)** which returns an array of the names of all provinces/territories whose name starts with letter (the parameter). Hint: use the String class’s **startsWith()** method or use the String class’s charAt(0) method.

This take-home lab is due next class. Finish it before next class (on your own…no partner) and bring it in person so your instructor can review it with you during the lab period after the lecture. Do not upload your lab to BCIT’s server.

Checked by: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**NOTE:** keep this paper for your instructor to verify your grades later in the course.

NOTE: EVERY SINGLE STUDENT MUST SUBMIT THIS LAB AND GET HIS OR HER PAPER SIGNED….