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

# Lab 7a: to be done in class with a partner – due at the end of this lesson

For this lab we will create a hierarchy of several classes: IDevice (the parent), IPod, IPad, and IPhone.

The parent class – IDevice – has a non-abstract method called **getPurpose()** which returns its **purpose** String. The **purpose** String instance variable is set in the constructor. The IDevice class also has an abstract method called printDetails() which prints out all of the child class’s instance variables.

Child classes of IDevice:

* IPod: the **purpose** of this iDevice is “music”
* IPad: the **purpose** of this iDevice is “learning”
* IPhone: the **purpose** of this iDevice is “talking”

Note: the child classes also contain instance variables, constructor parameters, accessor methods, and mutator methods for several other data members:

* IPod: (int) number of songs stored, (double) maximum volume in decibels
* IPad: (boolean) has a case, (String) operating system version
* IPhone: (double) number of minutes remaining on phone plan, (String) carrier

Also, each of these four classes also overrides toString() to return all of the instance variables in a String. Use the @Override annotation. Child classes’ toString() methods must also call their parent’s toString() method.

Furthermore, each of these four classes also overrides equals(). IPods with the same number of songs stored are considered equal; IPads with the same operating system version are considered equal; IPhones which have +/- 10 minutes of each other’s number of minutes remaining on their phone plan are considered equal. Use the @Override annotation.

Remember to override hashCode properly too for each class.

Demonstrate your completed project to your instructor. When your instructor is satisfied, your paper will be signed and you can go home. Lab 7b (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 7b: at home, alone – due next class (in person at the end of NEXT lab)

Continuing from lab 7a (above), add some data and methods as described below.

Extend the IPhone class; it has a child called IPhoneSeven.

The IPhoneSeven class also contains instance variables, constructor parameters, accessor methods, and mutator methods for several other data members:

* (boolean) high-resolution camera
* (int) gigabytes of memory

Also, the IPhoneSeven class overrides toString() to return all of the object data in a String. Use the @Override annotation. This toString() method must also call its parent’s toString() method.

Furthermore, this class also overrides equals(). IPhoneSevens that have +/- 10 minutes of each other’s number of minutes remaining on their phone plan are considered equal, but only if they also have the same value for “high-resolution camera”. Use the @Override annotation.

Remember to override hashCode properly too for this class.

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: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_