Mobile Computing

Lab 3

1 Statistics

1. Today we will be working with the android developer tools.

2. If you have not already set up android studio, please install it on your local machine.

1.1 In the MS Labs

If you are working in the MS Labs, you can set up android studio in your account by typing the following:

```
/opt/mc/reset
/opt/mc/setup
```

You can start Android Studio by opening a terminal and typing `/opt/mc/run`. You can click “Skip all and set defaults” in the resulting settings screen. When it prompts for a password, hit “Cancel”.

2 Getting started

1. When you have Android Studio running, create a new android project.

2. Call the project StatsApp

3. Depending on which version of android studio you have, you either need to set the company domain to `example.com` or set the package name to `com.example.statsapp`

4. Select “Empty Activity” in the “Project Template” selector.

5. Other than that, you can leave the default settings. Be careful, this step uses a lot of data.

6. This activity will be used to create a statistics calculator. It will allow the user to enter in a set of decimal numbers, and from these, calculate the mean, variance and standard deviation. Don’t worry if you haven’t ever studied stats, the math is reasonably straightforward.

3 Graphical User Interface - GUI

1. Go to the `res` folder, and from there go to the `layout` folder and open `activity_main.xml`.

2. This shows you the graphical layout of your page.

3. In order for your program to work, a user must be able to input numbers. In order to do this, there must be an edit box (Known on android as an EditText) on the page that can be used to enter decimal numbers. To prevent users from making mistakes, drag an EditText onto your app that ONLY allows decimal numbers. Change this EditText item's id property to editText1 in the Attributes pane on the right of the design window.

4. There must also be a button with the text `Add` so that the user can add the number they've typed to the list of numbers.

5. After they've entered their numbers, they must be able to ask for the mean, variance and standard deviation, so there must be buttons with the text `Mean`, `Variance` and `Std Dev`.

6. When they click on the buttons, we want to display the answer, so we need to drag a TextView onto the screen, giving it the text `Output`

4 Calculator

1. In your `src` folder, in the `com.example.statsapp` package, create a class called `Calculator` by right clicking on the package.

2. This class should contain static methods that all take in an ArrayList of Double objects, and return a `double`. These methods should be getMean, getVariance and getStdDev

3. Note that the ArrayList does not contain `double` variables, as these are primitive types. It instead contains `Double` objects, which are complex types, but you can work with them as if they are primitive types.

4. The getMean method should return the mean of all the numbers in the ArrayList. The formula for mean is given below.

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n}$$

   where μ is the mean.

5. The getVariance method should return the variance of all the numbers in the ArrayList. The formula for variance is given below.

$$\sigma^2 = \frac{\sum_{i=1}^{n} (x_i - \mu)^2}{n}$$

   where σ^2 is the variance of the numbers. Basically the variance is the sum of the squares of the difference between each number and the mean.

6. The getStdDev method should return the standard deviation of all the numbers in the ArrayList. The formula for standard deviation is given below.

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \mu)^2}{n}}$$

where σ is the standard deviation of the numbers. More simply, the standard deviation is the square root of the variance, or:

$$\sigma = \sqrt{\sigma^2}$$

5 MainActivity

1. Now we have to bind our interface to our calculator.

2. The first thing we need to do is define an onClick handler for each of our buttons. To do this, we edit `activity_main.xml`. To access the file, you need to either click on the `Text` tab at the bottom of the design window or the Code button on the top right of the design window, depending on which version of android studio you're using.

3. To tell android that when we click the add button we want to run a method called doAdd, we add the following line inside the xml for the add button:

   ```
   android:onClick="doAdd"
   ```

4. Now add onClick handlers to the Mean, Variance and Std Dev buttons to call `doMean`, `doVariance` and `doStdDev` methods respectively.

5. Now android knows to call these functions, but these functions do not exist yet. To create them, we edit the file `MainActivity.java`

6. In our MainActivity class, we must first create a function called `doAdd`. This method should look as follows:

   ```
   public void doAdd(View v){
     EditText e = (EditText)findViewById(R.id.editText1);
     String value = e.getText().toString();
     Double d = Double.parseDouble(value);
     numbers.add(d);
     e.setText("");
   }
   ```

7. This method gets called when the user clicks add. When that happens, the method gets the EditText that the user has been typing into, and gets the value out of it. This value is then converted into a Double and added to an ArrayList called `numbers`. Note that in order for this to work, you must create an `ArrayList<Double>` called `numbers` in your MainActivity class.

8. Now code the `doMean` method, which will take in a View in the same way that the doAdd method does. It must use the Calculator class to calculate the mean of the numbers in the `numbers` ArrayList, and must set the value of the Output TextView to the answer.

9. Now code the `doVariance` and `doStdDev` methods in the same way.

6 Submission

1. Submitting your code is a little different this week.
2. Your projects are stored in the a subfolder in your home directory called `AndroidStudioProjects`.
3. Go to the `app/src/main/java` folder of your project, then right click on the `com` subfolder, and zip it.
4. Then, submit this zip file to the marking system.