Name _____
Foundation of Programming

Miami Southridge High School
Bouwsma Rm 323

Project 7 Math Tutorial

9007210: 16.10,12,15, 17 .03, 05, 06, 11 18. 01, 02, 20. 02, 06, 21. 03, 04, 22 .04, 05, 23. 01-04, 24. 01, 04, 05, 25 .01, 04, 07

Background: Your neighbor's son, Tommy Gunn attends a private school named Dewy, Cheetim, and Howell High. Tommy's teacher, Cilia Fate, has sent home a failing progress report for math. Help your neighbor by creating a group of apps that will tutor Tommy in basic arithmetic. Tommy and his parents are depending upon you. (Five grades depend upon your work, as well.)

Part 1: Create Two Apps

Problem basics:

Tommy's low grades stem from the fact that **he is slow and inaccurate** when he adds, subtracts, multiplies, and divides. To get an **A**, your app needs to address both these problems. It needs to be easy to use, and easy on the eyes. It needs to be clear how well Tommy is doing and when Tommy's problem is fixed. Your app does not need to be fun. It should not have pretty pictures and internet access, or other things that would distract Tommy.

Procedure:

1. Choose a partner.
2. One partner needs to write an app for add and subtract, the other partner needs to make an app for multiply and divide.
3. The two programs need to have the same theme – that is, similar look, feel, coloring, and command structure.

Extras for extra credit:

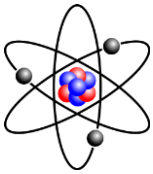
negative numbers, charts, graphs, usage history, printed reports, messages, instructional helps, lists of troublesome areas, and *especially your idea*.

Suggested Resources:

Explore, the class text book for Python programming. It contains instruction by example, techniques, and formulas. See https://bouwsma.neocities.org/docs/Explore.pdf

Notes on MIT's App Inventor 2. A tested compilation of block sequences that unlock the power of App Inventor 2. See https://bouwsma.neocities.org/docs/AppInventor2Notes.pdf

Multiply.exe, a PC program (not an app) that finds and fixes flaws in the user's knowledge of multiplication.



Feel Free To:

- Use the internet.
- Get help from those around you.
- Have friends test your work.

Demonstrate the completed app. **GRADE 1**

Part 2 Version 2.0

Make a second edition of your program.

1. Have **every student in the class** run your app at least once.
    a. Record their best score.
    b. Have one or more students use your app several times to measure its effectiveness.[1] Figure out how quickly they improve.
    c. Write up your results in a table and submit them. **GRADE 2**
2. Incorporate the best idea or ideas into your app. Here are the rules:
    a. The idea must not be your own, it must come from **user feedback**, following the Agile app development model.
    b. Your new feature needs to be a **clear improvement** to your program.
    c. **Credit** the user who gave you the idea somewhere in your app.
3. Show the instructor your upgrades. **GRADE 3**

Part 3 Deployment

It is time to publish your app following **GitJar** requirements (or Google Play Store requirements.[2])

1		**Unique App Name for listing**
2		**Developer name for listing** (avoid your own name)
3		**Short description ≤ 100[3] letters long.**
4		**Description ≤ 1,000[4] letters long.**
5		**Two to eight 480x800[5] px screen shots (Press Power & Vol Down to take a shot.)**

[1] Scientists call testing over a long period a longitudinal study. Usually it takes years, even decades for the study to be conclusive.
[2] If you elect to publish on Google Play Store, you receive extra credit on every grade, but it is far more difficult and far more public. It is particularly difficult to meet the privacy policy requirements if your app uses any of the phone's assets.
[3] Google Play Store allows only 80 letters and spaces.
[4] Google Play Store allows up to 4,000 letters and spaces.

6		**Three PNG pics for icons: 32x32, 64x64,128x128 px**[6]
7		**1024x500 JPG for a feature graphic picture that shows when user clicks to view your app.** (Google Play Store only)
8		**Privacy policy text.**
9		**Build name** (like Zit 3)
10		**Build Version** (like 1.0)
11		**Build Description** (features)
12		**Target** (usually Anyone)
13		**Short description** (of target)

Turn this table in for **GRADE 4.**

Part 4 Deployment

1. Make an account at **getjar.com**. Follow this link: : http://developer.getjar.mobi
2. Upload your pictures, information, and your **apk**[7].
3. Provide a **QR** on the packaging page you prepared earlier, for easy installation.

Your app will be used and tested by students in detention. **You will be graded by how much the students improve. GRADE 5**

Example pictures:

DO NOT USE ANY COPYRIGHTED TEXT OR PICTURES. ALWAYS CHECK PERMISSIONS BEFORE DOWNLOADING.

-- Icons -- ---------- Screen Shots ---------- ---------- Feature Graphic (Play Store) ----------

[5] Google Play Store allows a maximum dimension of 3840 px. The minimum size is 320 px. The other dimension must not be less than half the size of the largest

[6] Icons for Google Play Store must be 512x512.

[7] APK stands for Android PacKage. It is an executable file that will install on an Android phone.