mathe-tactics (**by cory fink)**

# project goal

The problem that my program addresses is students struggling to learn basic math through visual presentation. With this program, students are given hands-on access to learning math material through a series of input, outputs, and guided directions. This allows the teacher to teach 30+ students at the same time, through this guided program. It specifically focuses on students in Algebra 1, as the first lessons of the program are settled around lines and slopes. It allows the student to input their own information, which can be a point on a line, a slope, or a type of line they’d like to convert to slope-intercept form. This minimizes the typical textbook setting by allowing there to be user-program communication.

# MOTIVATION

My program is very valuable to others because the basis of the program is to *educate students*. It’s not for entertainment or for unpractical use, but to teach students with a different more technological approach to consuming information. Our world of education strives to use more adaptive skills through the practice of new technologies- this program does exactly that. It puts the computer to use by allowing students to learn new information at the touch of their fingertips. The program would also be available to the students at home, so practices can be used for homework, projects, or test prep. The program does not allow for any “non-sense” as the only time a user is inputting any information with their keyboard is for equation purposes, so only number contents are accepted.

# design and architecture

The program runs using a main runner, that “calls” other classes when buttons are clicked in the main runner. This then accesses that file and runs that specific “mini-program”. Each lesson is a mini-program that is implemented into the main runner, which can be considered the main console screen. This screen allows students to select specific lessons that they can work on. I used a tutorial which implements JFrame design through a hands-on console, which basically brings you to a design screen that creates the code for you behind the scenes. I still had to go back into the code to adjust things regarding Action Listeners, so the code still had to be developed and changed by myself. JFrame was the main topic of coding involved in this program, as well as buttons, action listeners, and scanner method. I implemented paint methods, action methods, JFrame painting methods, JPanel methods, as well as implementing a font class that allows you to use customized fonts for JTextAreas and JLabels. Very cool.

# implementation problems

The problems I had with JFrame were minimal, as it is a very creative coding technique and doesn’t require as much technical knowledge as it does ordering, designing, and planning knowledge. The problem I ran into, and still have, is allowing my mini programs to run in the same frame as my main console. When you click on a lesson off of the main console it “paints” a new JFrame and the program then takes place in that frame, this kind of takes away from the whole JFrame tactic, but I am still looking into improving this error. Another problem I ran into was adjusting the code after building it using the Design Panel thing that I downloaded. Problem was that when you would add anything to your frame using the Design Panel it would create the name for you, so going back into the code trying to locate a specific JLabel that you created or a JPanel was a little challenging. However the ctrl+F shortcut goes a long way.

# conclusion

I thought I achieved my goals set at the beginning of the project. My project started off with a simple Parallel line program run solely through the main console without any JFrames or any GUI output. It was all system.out.print( ) text. But after watching several tutorial methods and online forums Id say that I have a good knowledge of JFrame and GUI output, and look forward to using this to enhance my design and programming skills. I was very pleased that I could work on a project that would benefit me specifically in my future and help my own personal design skills. I was on task for majority of time with the exception of toying around with other programs in my Eclipse program to get a feel at how to go about certain aspects of my project. I was able to finish majority of the project in class, and only had to take it home to write my write-up and add some finishing touches to make the program run more smoothly.