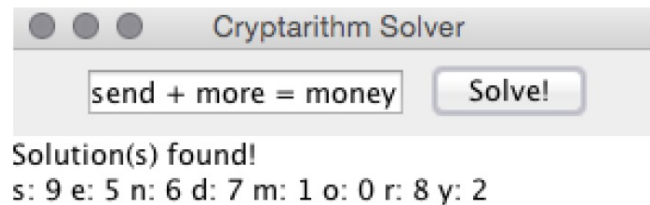


## Introduction to GUIs and Java Swing

This recitation will introduce you to several design patterns common within GUI applications and teach you some basics of Java Swing. Specifically, you will learn how to use Swing action listeners, layout managers, and how to retrieve information from text input fields. Your task is to complete the very simple Swing application described below.

### A GUI for your cryptarithm solver

You will complete a Swing application that will use a cryptarithm solver to solve user-entered cryptarithms. When you are done your application should look like:



This application contains three main parts: (1) a text field to input a cryptarithm, (2) a **Solve!** button that solves a cryptarithm, and (3) a text label where the result is shown.

We have provided some starter code in the 15-214 Recitation 07 project. To run the program you can run the `main` method in `Main.java`. We've provided a dummy cryptarithm solver you can use, or you can have your GUI use your own cryptarithm solver from Homework 3.

### Hints

1. `Main.java` and `CryptarithmSolverGui.java` contain detailed comments to help you start.
2. You are welcome to look for existing example Swing applications on the web.

### GUI design discussion

If you are successful, you should create a graphical user interface for a cryptarithm solver *without modifying the cryptarithm solver code*. This means that your GUI implementation depends on the cryptarithm solver, but the cryptarithm solver does not depend on your GUI implementation.

Why is it better for a GUI implementation to depend on some external components (such as your cryptarithm solver) independent of the GUI, rather than have the external components depend on the GUI? Use precise terminology and refer to design goals and design principles as possible.