TO: Jason^2, Manager

FROM: Jason, Manager

DATE: September 13, 2018

SUBJECT: Specification for Spreadsheet Package [Golang 1.11]

The spreadsheet will be structured as a rectangular arrangement of formulas. This will be represented by a struct containing a slice of slices, (henceforth adhering to the *sheet* interface) forming a two-dimensional grid, representing cells of the spreadsheet. These slices will be filled with structs that represent individual cells (henceforth adhering to the *formula* interface). This package should contain a pointer receiver for placing a *formula* into a specific cell of the spreadsheet. This function will take a set of coordinates that correspond to indices in the slice of slices to direct the *formula* to the proper place in the *sheet*. A *formula* should be able to contain a constant such as a string or a number, a reference to another cell or the addition and multiplication of two terms. We propose that it should have a receiver method *calculate()* which will return the evaluated value of this *formula*, taking references to other *formula* structs into account. We further propose that the struct have two values of type *formula*, A, and B, and a third value as an enumeration of values representative of ‘constant’, ‘multiplication,’ and ‘addition’. In the ‘constant’ case, the value will just be an float type cast of *formula* A. If it is one of the other values, it performs the according operation on the result of *calculate()* on A and B. We would also like a *new()* function for the construction of *formula* that takes A,B, and the enumeration value as arguments. It will return an error if this is not a valid configuration (A is not a float if ‘constant’ is used, A or B is nil if ‘multiplication’ or ‘addition’ is used, etc), and the formula struct if otherwise.

Sheet should also provide a receiver for determining the value of a cell. This function will take a set of coordinates to identify the cell to evaluate and then return the result of *calculate()* on that *formula*.

This interface should be implemented using Golang version 1.11.