From: Manager

To: Matthias^2

Subject: Spreadsheet program

Your task is to build a Spreadsheet class in Javascript, compatible with ES6 standards and the Node.js v6.9.4 environment. The deliverable is a module that exports a Spreadsheet class, which stores a grid of formatted numeric values, formulas, and cell references. The class should expose a constructor and two methods: one to set a cell's contents, and one to evaluate a cell.

The formal definition of the Spreadsheet data is as follows. A Spreadsheet is a rectangular 2D array of Formulas. A Formula is an Atom, an Expression, or the empty string. An Atom is a Number or a Reference to another cell. A Number is a numeric string. A Reference is a 2-part string that denotes a position in a spreadsheet. The column index is an all-uppercase alphabetic index, following standard spreadsheet conventions, and the row index is a positive integer. An Expression is a string composed of an Atom, an Operation, and a second Atom, in that order. An Operation is either '+' or '*', representing addition or multiplication.

The constructor for a Spreadsheet accepts a 2D array representing a list of the spreadsheet's rows, where each row is a list of Formula strings corresponding to the values in each column of that row. The Spreadsheet may not contain invalid references that point to cells outside of the Spreadsheet, or circular references where a cell's value depends on itself. This means that a Spreadsheet must be representable as a directed, acyclic graph, where cells are the nodes and references are the edges. Edges are directed from the cell that contains the reference to the cell indicated by the reference.

The evaluate() method must take a Reference to a cell and return the Number value of that cell, recursively evaluating its referenced cells and applying mathematical operations as indicated in any Expressions found. It should throw an error if the given Reference is invalid.

The set() method must take a Reference and a Formula, and will set the cell's contents to the given Formula. If the resulting spreadsheet would violate the validity constraints enforced by the constructor, or if either the Reference or the Formula are invalid, then set() does not update the cell and instead throws an error.