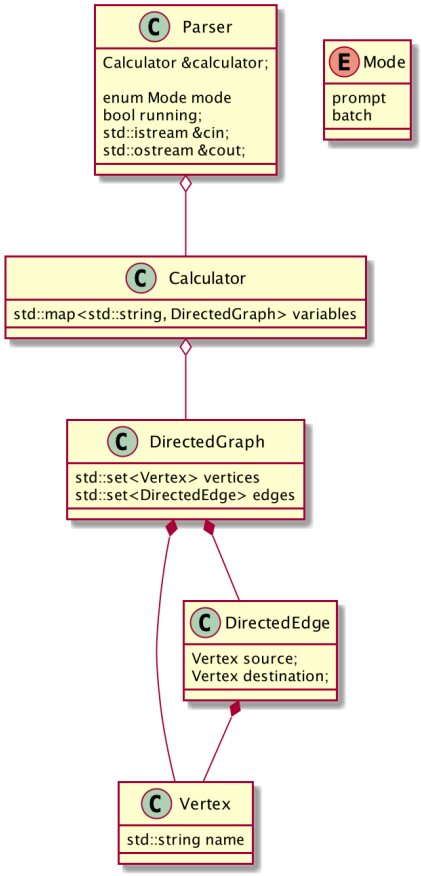
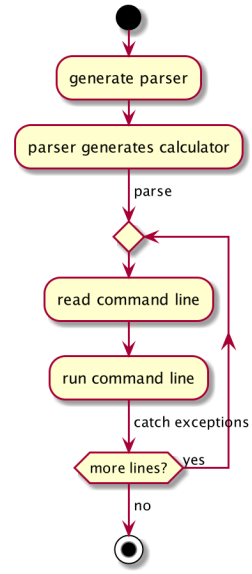
Matam Final Project – Roy Levi



Class Diagram

- Vertex has a name which it is identified by. name validity is verified as legal on creation.
- DirectedEdge is made out of a source and destination Vertex.
- DirectedGraph is made out of a set of vertices and a set of edges. Vertices and edges validity is verified as legal on creation.
- DirectedGraph also has functions for reading and writing to files.
- Calculator has a map of variable name to their corresponding DirectedGraph.
- Calculator can perform simple manipulations of its variables, such as printing, deleting, etc.
- Parser is responsible of interpreting command lines from a given file (or std::cin) and running them on a given Calculator, and printing output to a given file (or std::cout).



Run command line sequence

- Check type of command (assignment, save, print, reset, etc.) and evaluate DirectedGraph expressions as needed. For example: g = <Expression>.
- Evaluate expression:
    - Reads expression from left and split to variable stack and operators stack.
    - Graph literals, variables, and graph files go to the variables stack as DirectedGraph, and operators such as “!+^-*()” go to the operators stack
    - Each read operator triggers execution of previous operators on variables from the stack. For Example: in “g1+g2*g3” the operator * triggers the execution of +, and the operator) triggers all operators until it reaches (.
    - Return remaining DirectedGraph in variable stack after all operators are read and executed; throws exceptions as needed.
- Execute command line on calculator.