

*Note: The chart attached outlines a preliminary version of our Math Equation Solver project. That draft serves as a guarantee of the minimum functionalities we intend to implement. However, we are considering a range of additional features that could allow our program to handle more complex equations and mathematical operations.*

## **Ideas for Extra Classes and Functionality**

### **1) Vector Class**

instance variables

- vector (array of doubles)

methods

- the dot product of two vectors
- the cross product of two vectors (we can use the are calculation method for ex1 hw5)
- cosine of the angle formed by two vectors

### **2) GetDerivative Class**

methods

- find the derivative of the function at a given point (derivatives of polynomials, trig, and log functions)
- equation of a tangent line of the function at a given point a:  
 $f'(a) * (x-a) + f(a)$
- finding higher-order derivatives

### **3) SystemOfEquations Class**

instance variables

- LinearEquation and QuadraticEquation objects

methods

- Find points of intersection points of two lines, parabolas, or a parabola and line, considering them as solutions for the system of those equations
- given two equations of a line, determine if lines are parallel, intersect, or coincide

### **4) QuadraticEquationClass**

instance variables

- a, b, c for  $ax^2 + bx + c$  (coefficients)

methods

- Solve  $ax^2 + bx + c = 0$
- Solve  $ax^2 + bx + c = d$ , where d is a constant
- Finding the vertex of a parabola  $x_0 = -b/a$
- Number of solutions of the Equation  $ax^2 + bx + c = 0$   
(no solutions, one solution, two solutions)
- Discriminant Calculator

### **5) Implementation of trig and log functions**

- sin
- cos

- $\tan$
- $\cot$  (will use  $\tan$ )
- $\log$  (including  $\ln$ )