This project has to be implemented with as many concepts as possible that you have learned in the C++ module.
You will be implementing an equation interpretor, where the equation is given in std input in string format and has to be parsed and interpreted.

Your score will be proportional to the usage of concepts such as inheritance, polymorphism, and composition, and the usage of the standard template library. You will get a score for the group effort iin integrating the task combination for your group. Implementation and testing of your code independently as well as in the integrated mode are expected.

The integration of tasks will require code design involving different *types* of variables, e.g. coefficients (which are real numbers), unknowns (e.g. x, y, z), keywords (e.g. cos, sin), etc. In the integration of all tasks, the equation type must also be identified, before progressing to solve it.

Find the task descriptions below with examples and expected outputs. The output must be a string.
[*Note*: Your output does not need to have the double-quotes. Here they have been used to indicate that the output is a (concatenated) string. However, your input needs double-quotes to input strings. The use-case would be inputting an entire program code.]

Care must be taken for the following:
1. Interpret "a-b" as "a+(-b)".
2. Interpret "a+bx" as "bx+a".
3. Outputs with more than one value of the variable must be given in ascending order in left-to-right parsing.
4. Variable names must be strictly followed as per the description in the task. Thus, in all cases the output should be in the order of values of x, followed by y, and then z, as the task warrants.
5. All solutions must have decimal precision to 4 places, i.e. values with 4 decimal digits.

Task 1: Linear equations in one variable called "x".
$./myeqninterp "5x+6=20"
$ "x=2.8"

Task 2: Linear equations in two variables called "x" and "y", where the two equations are separated by ";".
$./myeqninterp "3x+5y=13;2x+3y=8"
$ "x=1;y=2"

Task 3: Linear equations in one variable called "x", with values in left- and right-hand sides of the equation.
$./myeqninterp "x+6=x+7"
$ "No solution"
$./myeqninterp "x+6=2x+27"
$ "x=-21"

Task 4: Quadratic equations in one variable called "x", where solutions "exist" in our case only if the discriminant is non-negative. The power operator is representing using the hat symbol "^".
$./myeqninterp "16x^2+8x+1=0"
$"x=0.25;x=0.25"
$./myeqninterp "x^2-16=0"
$"x=-4;x=4"
$./myeqninterp "x^2+x+1=0"
$"No solution"

Task 5: Non-homogeneous linear system equations with three variables called "x", "y", "z", which can have cases of unique, or non-unique (i.e. no, and infinite) solutions. The equations are separated using ";".
$./myeqninterp "x+y+z=1;2x+2y+2z=3;4x+4y+4z=3"
$"Non-unique solution"
$./myeqninterp "3x−2y+z=12;x+3y+z=−4;2x+2y−4z=6"
$"x=3;y=-2;z=-1"

Task 6: Trigonometric equation of the tangent form, with one variable called "x", where the solution is in radians. The equation format will be either "a*sin(x)=b*cos(x)" or "a*sin(x)-b*cos(x)=0".
$./myeqninterp "2sin(x)=3cos(x)"
$"0.9826"

Task 7: Exponential equation to be solved using natural logarithms, for one variable called "x". We use the hat "^" for the power operator. The equation is of type "a^(bx+c)=d^(ex+f)"
This equation is then solved as a linear equation of the form: ln(a).(bx+c)=ln(d).(ex+f)
$./myeqninterp "2^(4x+1)=3^(x+1)"
$"x=0.2422"

Task 8: Logarithmic equations with linear functions of variable called "x", of the format:
ln(ax+b)+ln(cx+d)=g.
This equation is then solved as a quadratic equation: (ax+b)(cx+d)=e^g
$./myeqninterp "ln(3x+4)+ln(2x+5)=2"
$"x=0.6630;x=3.1704"
$./myeqninterp "ln(x+1)+ln(x-1)=0"
$"x=-1.4142;x=1.4142"

Task 9: Absolute value equations using linear functions of the format "|ax+b|=cx+d", which reduces to two linear equations "ax+b=cx+d" and "-ax-b=cx+d", and solving them gives two values for the variable called "x".
$./myeqninterp "|x−2|=3x+1"
$"x=-1.5;x=0.25"
$./myeqninterp "|4x+3|=3−x"
$"x=-5;x=-1.3333"

[Credit: some of the equation types have been inspired from the list given in Paul's Online Notes in Algebra http://tutorial.math.lamar.edu/Classes/Alg/SolveAbsValueEqns.aspx]