**calExpression(char\* strIn, char\* strOut, char\* ans)**

strIn: input; strOut: output; ans: previous answer

Use recursive algorithm to separate input into small enough strings to calculate using math functions.

A valid input should be  
[<number> <operator> <number> <operator> … <number> ] or  
[<operator> <number> <operator> … <number>]

That is, it must start and end with a number. If it doesn’t start with a number, the first operator must be + or -

Operators are + - x / ^

Everything else are numbers: 12; 12.3; -0.134; 21!; <dấu căn>9; sin(1.2); ln(e); abs(-9.21); e; <pi>; ans; (-12.8);

What is in calExpression:

* Check for MATH ERROR or SYNTAX ERROR
* Looking for + -  
  If exist, addRealNum(calExpression(<left portion of input, as seen by operator>), calExpression(<left portion of input, as seen by operator>)); (or subtractRealNum)
* Looking for x /  
  If exist, do as + -
* Looking for ^  
  If exist, do as + -
* Looking for number  
  If exist, use math functions to calculate the number
* Return SYNTAX ERROR

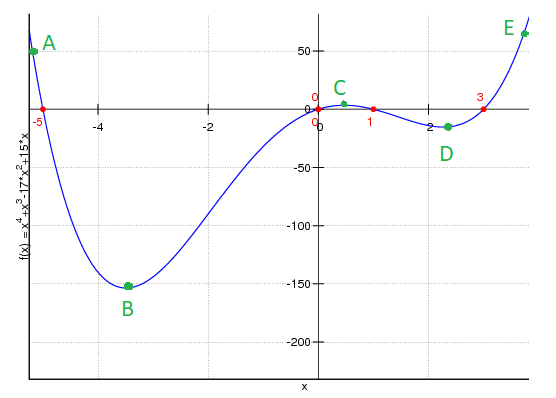
**Algorithm for solving an nth-degree equation:**

To solve an nth-degree equation, we find all the extreme points of the graph. Since the equation is always continuous, between 2 consecutive extreme points which is opposite in sign must lie a point where y = 0. Then we use binary search to find the solution with a certain accuracy. In practice, we also consider (- infinity; y) and (infinity; y) extreme points.

To find the extreme points of the graph, we need to solve the equation of the derivative of the nth-degree equation, which is an (n-1)th-degree equation. This lead to a recursive algorithm, which stops when n = 1.

For example:

Solve



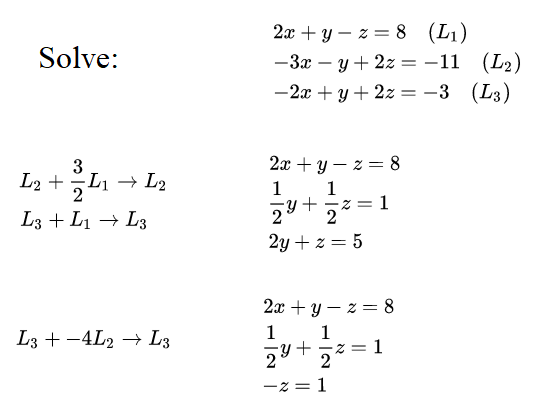
Solving y’ = 0, we get B, C, D. As mentioned before, (- infinity; y) and (infinity; y) are also considered extreme points so we add A and E to the group of extreme points.

As you can see, , between 2 consecutive extreme points which is opposite in sign, there always is a point where y = 0, so we use binary search between those 2 points to find the solution.

Disavantage: As the datatype double is not accurate, the program may miss solutions with big absolute value.

**Algorithm for solving a set of n equations:**

Using Gaussian elimination (phép khử Gauss)



Thôi m để hình thôi rồi t nói chứ quăng nguyên cái thuật toán lên slide cũng ko ổn :v

**room.Solve:**

* Xử lí trên string
* Đổi qua lại giữa các hệ 10 2 8 16
* Đổi được số nguyên, số thập phân âm dương

**room.Setting:**

* Đổi màu chữ
* Đổi cỡ chữ
* Lưu màu chữ và cỡ chữ vào file

**room.Dinosaur:** for fun =)))))