**Overloading Operator**

***BY: Nathan almaleh 209633635 && Adam Lev Ari 204115547***

Purpose of this assignment is Overloads operator.  
  
solve((x ^ 2) + 6 \* x == -9);

cout << x << endl;  
  
By typing a linear expression in the main as you can see.  
We wish to output the solution.  
Doing so we need to overload all the operators so the expression will be acceptable.  
 ***Project Classes and Function explained.***

Classes:  
 class Var  
 class Poly  
 class Line   
 class OperatorShift  
 class Demo   
 class Badkan

Class Var:  
 float i; // I contain the number

Class Line: //a\*x+b or a\*x^c+b  
 float a, b; //numbers

VAR\* x; //pointer to the struct X

LINE() {} //default constructor

LINE(int b) : a(0), x(0), b(b) {} //constructor with number in expression

LINE(VAR& v) : a(1), x(&v), b(0) {} //constructor for number next to the X

Class Poly: //ax^e+bx+c

float a, b, c; //numbers

int e; //power

VAR\* x; //pointer to the struct X

POLI() {} //default constructor

POLI(float a, int e) : a(a) {} //constructor with number close to X

Class OperatorShift: //contains all the operators for Line and Poly  
 POLI operator ^ (LINE A, int e) //power operator  
 LINE operator + (LINE A, LINE B) // + operator  
 POLI operator + (POLI A, LINE B) // + operator  
 LINE operator - (LINE A, LINE B) // - operator  
 POLI operator - (POLI A, LINE B) // - operator  
 LINE operator \* (LINE A, LINE B) // \* operator  
 LINE operator / (LINE A, LINE B) // / operator  
 LINE operator == (LINE A, LINE B) // == operator  
 POLI operator == (POLI A, LINE B) // == operator  
 ostream &operator << (ostream &output, const VAR &A) //output operator  
 string cast(POLI x)   
 int solve(POLI x) //returns value of X  
 int solve(LINE x) //returns value of X

Class Demo:  
 int main(){  
 VAR x;   
 solve((2\*x^2) - 4\*x + 3 == 9);  
 cout << x << endl; //Output 3  
 solve((x^2) + 6\*x == -9);  
 cout << x << endl; //Output -3  
 solve(2\*x-4 == 10);   
 cout << x << endl; //Output 7  
 return 0;  
 }