Parser

protected string[] tokens
protected int pos
protected Stack<int> memory
protected bool debug {get;set;}
protected Stack<ASTree> forest
protected Stack<int> forestMemory
public static ASTree LastExpression

---

public static bool Parse(string expr, bool debug = false)
public bool ParseCmds(string expr, bool debug = false)
protected void d(string s)
protected string token()
 protected void nextToken()
protected void memorize()
 protected void recall()
protected bool eos()
protected bool script()
protected bool end_script()
protected bool affect_expr()
protected bool affect_operator()
protected bool end_affect_expr(bool isVar)
protected bool arith_expr()
protected bool end_arith_expr(bool isTerm)
protected bool term()
 protected bool isOperator()
protected bool end_term(bool isFactor)
protected bool unaryMinus()
protected bool end_unary_minus()
protected bool factor()
protected bool variable()
protected bool numeric()
protected bool function()
 protected bool end_function(bool isArgList)
protected bool func_name()
protected bool arg_list()
protected bool end_arg_list(bool isArg)
public static void SendMsg(string expr, bool debug = false)

Index

public static void Main()
public static void DisplayTests()

ASTree

public string Root
public ASType Type
public List<ASTree> Children

---

public void AddChild(ASTree t)
public double Evaluate(Dictionary<string,double>m)
public string Serialize(IExporter exporter)
public override string ToString()

Calculateur

public enum ASType{}

public interface IExporter

string Export(ASTree t)

public class LispExporter

string Export(ASTree t)

public class RPN

string Export(ASTree t)

public class TextExporter

string Export(ASTree t)