Cse 222 HW05 Report

Okan Akdogan

121044017

HW4 Report

Requirements

Inputs: a file that contains expressions

A number when input operator run

Output: console out expression results

Problem Analysis and Design

Program takes a file (ex: file.git) and reads it line by line.every line represents an expression.

In main using File.ReadAllLine method reading file and assign the String array.

There are two classes for manipulate and solve these expressions in array.

The InfixToPostFix class has a convert method for rewrite expression in postfix

The PostfixEvulator class has a eval method for evaluate postfix expression.

Also there is a List of Variables that contains in expressions declared in main. For this List, there is a Variable class that includes fields variable name and its value.

Classes

Class InfixToPostFix()

Fields

Final String[] OPERATORS – Operator list

Stack<String> operatorStack – stack for opeartor that read in expression

StringBuilder postfix – postfix expression will be in this string

Final İNT[] PRECEDENCE – precedence of operators

Methods

Public convert – converts infix expression in string to postfix expression

isOperator – checks given string is it operator or not

precedence – gives precedence number of operators

proccesOperator – creates postfix expression in rules of precedence

Class PostFixEvaluator()

Fields

Final String[] OPERATORS – Operator list

Final int[] OPERATORTYPE – number of operand that operator takes

LinkedList<Variable> Variables – variable stack for variables that declared in expressions

Stack<String> operandStack – stack for operands that read in expression

Methods

addVariable – adds varibale to Variables stack if not exist

getvar – gets variable from stack

isContainVar- checks variable is exist in Variables stack

eval - evaluates given string of postfix expression with given Variable list

evalOp - makes operate for given operators with operands

isOperator – checks given string is it operator or not

operatorType – gives type of operator, 1 if unary ,2 if binary

class Variable()

Fields

varName- variable name

value – variable value

Methods

getName- gets varName

getValue- gets value

setName – sets varName

setValue – sets value

Variable – empty or parameted constructors

News in HW5

In main method program handles "loop" ,"begin" and "end" keywords in .git files. Using a LinkedList of LinkedList<Variable> to hold all variables seperate for each loop. First Variable list holds main program variables. Other linked lists are created with loop keyword. With this method if variable declartion is in loop (local variable) ,variable will be in loop Variable list. When loop done this List will be delete.

Program can takes nested loops.