

Group Name:

Section:

Member 1: FRIAS, Mon Cedrick

Member 3: MENGUITO, James Paolo

Member 2: MARCOS, Joseph Matthew

Member 4:

Identifiers

PL Element	Regular Expression
Variable Identifier	<code>^[a-zA-Z] [a-zA-Z\d]*\$</code>
Function Identifier	<code>^[a-zA-Z] [a-zA-Z\d]*\$</code>
Loop Identifier	<code>^[a-zA-Z] [a-zA-Z\d]*\$</code>

Literals

PL Element	Regular Expression
NUMBR Literal	<code>^\-?\d+\$</code>
NUMBAR Literal	<code>^\-?\d*\.\d+\$</code>
YARN Literal	<code>^"(\s \w)*"\$</code>
TROOF Literal	<code>^(WIN FAIL)\$</code>
TYPE Literal	<code>^(TROOF YARN NUMBR NUMBAR NOOB)\$</code>

Keywords

PL Element	When/how to use?	Regular Expression
HAI	Start/opens the program	<code>^HAI\$</code>
KTHXBYE	End/closes the program	<code>^KTHXBAI\$</code>
BTW	Single line comment	<code>^BTW\$</code>
OBTW	Start of multi-line comment	<code>^OBTW\$</code>
TLDR	End of multi-line comment	<code>^TLDR\$</code>
I HAS A	Declares a variable followed by a variable name	<code>^I[]HAS[]A\$</code>
ITZ	Assigns a value within the same statement, follows after the variable name	<code>^ITZ\$</code>
R	Assigns a variable together with an assignment statement	<code>^R\$</code>
SUM OF	Addition operator	<code>^SUM[]OF\$</code>
DIFF OF	Subtraction operator	<code>^DIFF[]OF\$</code>
PRODUKT OF	Product operator	<code>^PRODUKT[]OF\$</code>

Design and Implementation of Programming Languages

Regex

QUOSHUNT OF	Quotient operator	^QUOSHUNT[]OF\$
MOD OF	Modulo operator	^MOD[]OF\$
BIGGR OF	Returns the maximum value	^BIGGR[]OF\$
SMALLR OF	Returns the minimum value	^SMALLR[]OF\$
BOTH OF	AND operator	^BOTH[]OF\$
EITHER OF	OR operator	^EITHER[]OF\$
WON OF	XOR operator	^WON[]OF\$
NOT	Negation operator	^NOT\$
ALL OF	Infinite arity AND	^ALL[]OF\$
ANY OF	Infinite arity OR	^ANY[]OF\$
BOTH SAEM	Compares the two operations and returns WIN (true) iff op1 == op2	^BOTH[]SAEM\$
DIFFRINT	Compares the two operations and returns WIN (true) iff op1 != op2	^DIFFRINT\$
SMOOSH	String (YARN) concatenation operator	^SMOOSH\$
MAEK	Casting assignment statement	^MAEK\$
A	Found between a variable and a type in casting assignment statement, especially in make operator	^A\$
IS NOW A	Casting assignment statement	^IS[]NOW[]A\$
VISIBLE	Print operator	^VISIBLE\$
GIMMEH	Accepts input from the user	^GIMMEH\$
O RLY?	Start of If-then condition	^O[]RLY? \$
YA RLY	If; Go or do this block if IT is cast to WIN	^YA[]RLY\$
MEBBE	Elseif; may appear between YA RLY and NO WAI blocks, go or do this block if the expression following MEBBE is WIN	^MEBBE\$
NO WAI	Else; Go or do this block if IT is cast to FAIL	^NO[]WAI\$

Design and Implementation of Programming Languages

Regex

OIC	End of If-then and Case condition	^OIC\$
WTF?	Start of switch statements	^WTF\?\$
OMG	Case(s) of switch statements, except the default case	^OMG\$
OMGWTF	Optional default case	^OMGWTF\$
IM IN YR	Start of loop	^IM[]IN[]YR\$
UPPIN	Increment a value by 1	^UPPIN\$
NERFIN	Decrement a value by 1	^NERFIN\$
YR	Found between an operation and a variable in loops	^YR\$
TIL	If the expression is FAIL, continue the execution of the loop. Otherwise, stop.	^TIL\$
WILE	If the expression is WIN, continue the execution of the loop. Otherwise, stop.	^WILE\$
IM OUTTA YR	End of loop	^IM[]OUTTA[]YR\$