

Group Name: Team WLMB	Section: T5L
Member 1: Dayahan, Rey Russel M.	
Member 2: Gines, Kevin A.	

LOLCODE GRAMMAR

Phrases enclosed by angle brackets (<,>) are abstractions. **Words in small letters describe the lexemes that are already described by a regular expression** (e.g. varident for variable identifiers, yarn for string literals, troof for boolean values, etc).

LHS	::=	RHS
<program>	::=	HAI <linebreak> <statement> <linebreak> KTHXBYE
<statement>	::=	<print> <linebreak> <vardec> <input> <assignment> <ifelse> <switchcase> <loop> <function> <functioncall> <statement>
<codeblock>	::=	<print> <linebreak> <expr> <assignment> <ifelse> <vardec> <switchcase> <loop> <functioncall> <codeblock>
<print>	::=	VISIBLE <visibleoperand>
<visibleoperand>	::=	yarn <expr> <literal>
<expr>	::=	<arithmetic> <boolean> <comparison>
<literal>	::=	numbr numbar yarn troof
<linebreak>	::=	\n
<input>	::=	GIMMEH varident
<vardec>	::=	I HAS A varident I HAS A varident <varinit>
<varinit>	::=	ITZ <literal> ITZ <expr> ITZ varident ITZ <typecast>
<arithmetic>	::=	<operation1> <operand> [AN] <operand>
<operation1>	::=	SUM OF DIFF OF BIGGR OF SMALLR OF PRODUKT OF QUOSHUNT MOD OF
<operation2>	::=	BIGGR OF SMALLR OF
<operand>	::=	varident numbr numbar <arithmetic>
<boolean>	::=	<booloperation1> <booloperand1> [AN] <booloperand1> <unary> <booloperand1> <boolean2>
<boolean2>	::=	<booloperation2> <booloperand2> [AN] <booloperand2> MKAY
<booloperation1>	::=	BOTH OF EITHER OF WON OF
<booloperation2>	::=	ANY OF ALL OF
<booloperand1>	::=	varident troof <boolean>
<booloperand2>	::=	varident troof <booleanf2only> varident AN <booloperand2> troof AN <booloperand2> <booleanf2only> AN <booloperand2>
<booleanf2only>	::=	<booloperation1> <booloperand2> [AN] <booloperand2> NOT <booloperand2>

<comparison>	::=	<compoperator> <operand> AN <operand> <compoperator> <operand> <operation2> <operand>
<compoperator>	::=	BOTH SAEM DIFFRINT
<concatenation>	::=	SMOOSH <concatenand> [AN] <concatenand>
<concatenand>	::=	yarn [AN] <concatenand> yarn
<typecast>	::=	MAEK varident A <literal> MAEK <literal> A <literal>
<assignment>	::=	varident R <typecast> varident R varident varident R <literal> varident R <expr> varident R <concatenation>
<ifcond>	::=	<boolean> <comparison>
<ifelse>	::=	<ifcond> <linebreak> 0 RLY? <ifclause> <elseclause> 0IC <ifcond> 0 RLY? <ifclause><mebbeclause><elseclause> 0IC
<ifclause>	::=	<linebreak> YA RLY <linebreak><codeblock><linebreak>
<elseclause>	::=	<linebreak> NO WAI <linebreak><codeblock><linebreak>
<mebbeclause>	::=	<linebreak> MEBBE <ifcond><linebreak><codeblock><linebreak> <linebreak> MEBBE <ifcond><linebreak><codeblock><linebreak><mebbeclause>
<switchcase>	::=	WTF? <linebreak> <case> 0IC
<case>	::=	OMG <literal><linebreak><codeblock><linebreak>GTF0<linebreak><case> OMGWTF <linebreak><codeblock><linebreak>
<loop>	::=	IM IN YR loopident <unary> YR varident <looptype><ifcond><linebreak><codeblock><linebreak> IM OUTTA YR loopident IM IN YR loopident <unary> YR varident <linebreak><codeblock><linebreak> GTF0 <linebreak> IM OUTTA YR loopident
<unary>	::=	UPPIN NERFIN NOT
<looptype>	::=	TIL WILE
<function>	::=	HOW IZ I fxnident YR <fxnparameter> <linebreak><codeblock><linebreak><returnstatement> HOW IZ I fxnident<linebreak> <codeblock><linebreak><returnstatement>
<fxnparameter>	::=	varident AN <fxnparameter> varident
<returnstatement>	::=	IF U SAY SO FOUND YR <returnvalue> GTF0
<returnvalue>	::=	varident <expr> literal
<fxnargument>	::=	<returnvalue> AN <fxnargument> <returnvalue>
<functioncall>	::=	I IZ fxnident <fxnargument> MKAY I IZ fxnident MKAY
<comment>	::=	BTW yarn <linebreak><OBTW> yarn <linebreak> TLDR <linebreak>