CMSC 124 Design and Implementation of Programming Languages **First Semester AY 17-18**

LOLCODE Grammar **Project Requirement 02**

Group Name: Dobol Cobol	**Section: T-1L**
Member 1: John Gabriel Bayta	**Member 3:**
Member 2: Akemi Ray Goshi	**Member 4:**

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>	::=	<statement><statement><linebreak> | <expression> | <switchcase> | <ifthen> | <varcall> | <print> | <input> | <comments>
<print>	::=	VISIBLE varident | VISIBLE <expr> | VISIBLE <literal>
<varcall>	::=	I HAS A varident | I HAS A varident ITZ <literal> | I HAS A varident ITZ <expression>
<comments>	::=	BTW comment | OBTW <linebreak> comment <linebreak> TLDR
<expression>	::=	<comparisonop> | <arithmeticop> | <booleanop> | <concatop> | <assignmentop>
<arithmeticop>	::=	SUM OF <arithmeticop> <litvar> AN <litvar> | DIFF OF <arithmeticop> <litvar> AN <litvar> | PRODUKT OF <arithmeticop> <litvar> AN <litvar> | QUOSHUNT OF <arithmeticop> <litvar> AN <litvar> | MOD OF <arithmeticop> <litvar> AN <litvar> | BIGGR OF <arithmeticop> <litvar> AN <litvar> | SMALLR OF <arithmeticop> <litvar> AN <litvar> | SUM OF <litvar> AN <litvar> | DIFF OF <litvar> AN <litvar> | PRODUKT OF <litvar> AN <litvar> | QUOSHUNT OF <litvar> AN <litvar> | MOD OF <litvar> AN <litvar> | BIGGR OF <litvar> AN <litvar> | SMALLR OF <litvar> AN <litvar>
<booleanop>	::=	BOTH OF <litvar> AN <litvar> | EITHER OF <litvar> AN <litvar> | WON OF <litvar> AN <litvar> | NOT <litvar> | ALL OF <boolean> AN <boolean> ...MKAY | ALL OF <litvar> AN <litvar> ...MKAY | ANY OF <booleanclause> ...MKAY | ANY OF <litvar> AN <litvar> ...MKAY
<comparisonop>	::=	BOTH SAEM <litvar> AN <litvar> | DIFFRINT <litvar> AN <litvar> | BOTH SAEM <litvar> AN BIGGR OF <litvar> AN <litvar> | BOTH SAEM <litvar> AN SMALLR OF <litvar> AN <litvar> | DIFFRINT <litvar> AN BIGGR OF <litvar> AN <litvar> | DIFFRINT <litvar> AN SMALLR OF <litvar> AN <litvar>
<concatop>	::=	SMOOSH <anclause>
<input>	::=	GIMMEH varident
<assignment>	::=	varident R <literal> | varident R varident | varident R <expression>
<ifthen>	::=	<expression><linebreak>O RLY?<linebreak> YA RLY<linebreak><statement><linebreak> NO WAI<linebreak><statement><linebreak> OIC

<switchcase>	::=	WTF? <linebreak><omgclause> OIC| WTF? <linebreak><omgclause> OMGWTF <linebreak><statement> OIC
<anclause>	::=	<literal> AN <literal> | <anclause><anclause>
<booleanclause>	::=	<boolean> AN <boolean> | <boolean><boolean> AN <boolean><boolean>
<loops>	::=	IM IN YR <label> <loopop> YR varident <testexpression><linebreak><statement><linebreak> IM OUTTA YR <label>
<omgclause>	::=	OMG <literal><linebreak><statement> | <omgclause><omgclause>
<testexpression>	::=	TIL <expression> | WILE <expression>
<loopop>	::=	UPPIN | NERFIN
<litvar>	::=	<literal> | varident
<literal>	::=	numbr | numbar | yarn | <troof>
<troof>	::=	WIN | FAIL