%{

#include <string>

#include <iostream>

using namespace std;

int commentLevel = 0;

%}

%x IN\_COMMENT

/\*Definitions of Some Basic Tokens \*/

INT [0-9]+

FLOAT 0"."[0-9]+

CHAR [a-zA-Z\-\\\\*]

%%

<INITIAL>("d"|"D")"ear "("k"|"K")"akubo"[,]? printf("<MAIN>");

<INITIAL>("p"|"P")"lease"[,]? printf("<BEGIN\_OF\_BLOCK>");

<INITIAL>("t"|"T")"hank you"["."]? printf("<END\_OF\_BLOCK>");

<INITIAL>[\!] printf("<END\_OF\_PRGM>");

<INITIAL>"." printf("<END\_OF\_INPUT>");

<INITIAL>("b"|"B")"est "("w"|"W")"ishes"["."]? printf("<END\_OF\_MAIN>");

<INITIAL>"P.S." printf("<FUNCTION>");

<INITIAL>"P.P.S." printf("<USR\_DEF\_SHAPE>");

<INITIAL>("I” | “i”)” defined the function" printf("<FUNCT\_NAME>");

<INITIAL>"as the following:" printf("<END\_OF\_DEF>");

<INITIAL>"which is a" printf("<INHERITANCE>");

<INITIAL>("I"|"i")" defined" printf("<USR\_SHAPE\_NAME>");

<INITIAL>("d"|"D")"raw" printf("<DRAW>");

<INITIAL>"position" printf("<POSITION>");

<INITIAL>"width" printf("<WIDTH>");

<INITIAL>"height" printf("<HEIGHT>");

<INITIAL>"stroke" printf("<STROKE>");

<INITIAL>"color" printf("<COLOR>");

<INITIAL>"fill state" printf("<FILL\_STATE>");

<INITIAL>"fill color" printf("<FILL\_COLOR>");

<INITIAL>"border state" printf("<BORDR\_STATE>");

<INITIAL>"border color" printf("<BORDR\_COLR>");

<INITIAL>"red" printf("<RED>");

<INITIAL>"blue" printf("<BLUE>");

<INITIAL>"green" printf("<GREEN>");

<INITIAL>"yellow" printf("<YELLOW>");

<INITIAL>"black" printf("<BLACK>");

<INITIAL>"white" printf("<WHITE>");

<INITIAL>“rounded corners” printf(“<RND\_CRNRS>”);

<INITIAL>"name" printf("<NAME>");

<INITIAL>”arrow start” printf(“<ARROW\_STRT>”);

<INITIAL>”arrow end” printf(“<ARROW\_END>”);

<INITIAL>("o"|"O")"val"|("r"|"R")"ectangle"|("l"|"L")"ine" printf("<SHAPE>");

<INITIAL>“direction” printf(“<DIR>”);

<INITIAL>“N”|”W”|”S”|”E”|”NW”|”NE”|”SW”|”SE” printf(“<DIR\_VAL”);

<INITIAL>“~”(”character”| ”char” | “integer” | “int”)”~” printf(“<TYPECAST>”);

<INITIAL>"if"|"with the condition that"|"in case of"|"in case that"|"on the occasion" printf("<IF>");

<INITIAL>"else"|"if not" printf("<ELSE>");

<INITIAL>[\+|\-|\\*|\/|\%]|"add"|"plus"|"minus"|"subtract"|"multiply"|"divide"|”modulo” printf("<OP>");

<INITIAL>("greater"|"bigger"|"larger")"than"|[>] printf("<GRTR>");

<INITIAL>("smaller"|"less")"than"|[<] printf("<LESS>");

<INITIAL>[=]|”is”|”assign” printf("<ASSIGNMENT>");

<INITIAL>“is equal to”|[==] printf(“<LOG\_EQ>”);

<INITIAL>”is “("greater"|"bigger"|"larger")" than or equal to"|">=" printf("<GRTR\_EQ>");

<INITIAL>”is “("less"|"smaller")" than or equal to"|"<=" printf("<LESS\_EQ>");

<INITIAL>"&&"|"and" printf("<AND>");

<INITIAL>"||"|"or" printf("<OR>");

<INITIAL>"times" printf("<FOR>");

<INITIAL>(“U”|”u”)”ntil” printf(“<WHILE>”);

<INITIAL>"["({INT}|{FLOAT})"]" printf("<NUM>");

<INITIAL>\"([^\"\\\n]|\\\"|\\n|\\t|\\\\)\*\" printf(“<STRING>”);

<INITIAL>\’{CHAR}\’ printf(“<CHAR>”);

<INITIAL>"," printf("<PARAMETER>");

<INITIAL>\[{NUM}“,”{NUM}“,”{NUM}\] printf(“<RGB>”);

<INITIAL>\[{NUM}“,”{NUM}\] printf(“<POS\_X\_POS\_Y>”);

<INITIAL>"with" printf(“<USR\_DEF\_PARAM>”);

<INITIAL>“‘s” printf(“<ACCESS>”);

<INITIAL>”has” printf(“<USR\_DEF\_VAR”);

<INITIAL>"at"|"a " | “an “ | ”using” ;

<INITIAL>"(" {

BEGIN(IN\_COMMENT);

level = 1;

}

<IN\_COMMENT>")" {

commentLevel--;

if (commentLevel==0)

BEGIN(INITIAL);

}

<IN\_COMMENT>"(" {

commentLevel++;

}

<INITIAL>[a-zA-Z\-\\\\*]+ printf("<ID>");

<INITIAL>[ \t\n]+ ;

%%

int yywrap(void) {

return 1;

}

int main(void) {

yylex();

return 0;

}

Colors need to be added: red, blue, green, white, black, yellow done

Rounded corners

Has a

Direction

Arrow

“ “

~char | int~

‘ ’

[ , ] -> postion

[ , , ] -> RGB

Assignment , is, equals to done

‘S

With

User defined variables

Modulo