**Antonio Cruz (adc2104)**
PLT Fall 2007
Professor Edwards
Language Reference Manual

## "flash-tree" (a.k.a. "SFAL" - Simple Flash Animation Language)

**SFAL** is a language designed for programming simple vector-graphic animation using Adobe Flash technology.

SFAL replaces my original language idea, "flash-tree".

## SFAL Language Reference Manual

### 1. Tokens

SFAL tokens consist of: identifiers, keywords, constants, string literals and separators.

### 2. Comments

SFAL comments may be single line comments beginning with //, or multi-line comments beginning with /* and ending with */.

### 3. Keywords

The following identifiers are reserved for use as keywords and may not be used otherwise:

```
makestage
makecircle
makeellipse
makerectangle
makesquare
makegroup
maketext
addtogroup

drawline
drawcurve

point
rotate
move
place
stop
bounce
reverse

sin
cos

setcolor
red
yellow
green
blue
brown
black
orange
purple
gray
```

### 4. Constants

SFAL supports integer constants and floating-point constants.

### 5. String Literals

SFAL supports string literals, also called string constants, which are sequences of characters enclosed by double quotes, e.g. "Hello World".

### 6. Assignment Expression

An assignment expression takes the form

*ID = (makecircle*
*| makeellipse*
*| makerectangle*
*| makesquare*
*| makegroup*
*| maketext) (arg | (arg,)+ arg) ;*

### 7. Statements

Statements end with a semicolon and may take the form

*assignment-expression or*
*function*

### 8. Functions

SFAL includes a number of built-in functions which operate on or return objects.

*function-name arg ; or*
*function-name (arg,)+ arg ;*

**Example code animating a group of graphic objects in a circle.**

```
// make a 500 by 500 pixel drawing stage
makestage 500, 500;

// make a blue square with sides 20 pixels long
square = makesquare 20;
setcolor square, blue;

// put the square at point 100, 100 on the stage
p2 = point 100, 100;
place square, p2;

// make a red circle with a radius of 20
circle = makecircle 20;
setcolor circle, red;

// put the circle at point 100, 120 on the stage
p2 = point 100, 120;
place circle, p1;

// group the circle and square together
circlesquare = makegroup;
addtogroup circlesquare, circle;
addtogroup circlesquare, square;

// rotate the circlesquare group
spin circlesquare 100, 100, .15;
```