easel
transforming math into art
The Team

- Danielle Crosswell - Manager
- Tyrus Cukavac - Language Guru
- Yuan-Chao (Oswin) Chou - System Architect
- Xiaofei (Sophie) Chen - Tester
The Goal

Take mathematical functions and turn them into images

Source: Manuel Kasten’s posting on http://codegolf.stackexchange.com/questions/35569/tweetable-mathematical-art
Architectural Design

Scanner

Parser

Semantic Checker

Code Generator

Abstract Syntax Tree (in memory)

Tokens (in memory)

easel source code (.es)

LLVM IR (.ll)

Executable

Object

Assembly (.s)

gcc, ar

llc

libglut.so

libGL.so

libglwrap.a

easel standard library (glwrap.c)

llc

gcc, ar
Language Syntax

/* Types */
int
float
bool
pix
array
matrix
function

/* Operators */
+ - * / ^ %
< > <= >= == !=
++ ! ++ --
&& ||

/* Control Flow */
if (true) {
    do something;
}
else {
    do something else;
}
do something;

/* Function declaration */
function int test(int a) {
    do something;
    return a;
}
function pix (bool b) { return !b; }

/* Built-in Functions */
draw(canv[][], x, y);
draw_size(canv[][], x, y, a, b);
sin(float f);
cos(float f);
tan(float f);
log(float b, float v);
rando();
randos(int s);
```javascript
/* squares.es */

/* sample easel program to return a checkered pattern */

pix col1 = #BF8FD608;
pix col2 = {70, 58, 39, 170};
pix canvas[960][720];

int W = 960;
int H = 720;

function void plaid(int p_w) {
    int x,y;
    for (y = 0; y < H; y++) {
        for (x = 0; x < W; x++) {
            if (((x / p_w) % 2) == ((y / p_w) % 2))
                canvas[x][y] = col1;
            else
                canvas[x][y] = col2;
        }
    }
    plaid(100);
draw(canvas, 0, 0);
}
Beautiful equations...?

```c
float a = 0., b = 0., c, d, n = 0.;
while ((c = a * a) + (d = b * b) < 4. && n++ < 880.) {
    b = 2. * a * b + y * 8e-9 - .645411;
    a = c - d + x * 8e-9 + .356888;
}
return 255 * (((n - 80.)/800.) ^ 3.);

float a = 0., b = 0., c, d, n = 0.;
while ((c = a * a) + (d = b * b) < 4. && n++ < 880.) {
    b = 2. * a * b + y * 8e-9 - .645411;
    a = c - d + x * 8e-9 + .356888;
}
return 255 * (((n - 80.)/800.) ^ .7);

float a = 0., b = 0., c, d, n = 0.;
while ((c = a * a) + (d = b * b) < 4. && n++ < 880.) {
    b = 2. * a * b + y * 8e-9 - .645411;
    a = c - d + x * 8e-9 + .356888;
}
return 255 * (((n - 80.)/800.) ^ .5);
```
easel’s got you covered!

```javascript
function void graph(pix[960][960] canv, int w, int h, function pix (int, int) painter) {
    int x, y;
    for (y = 0; y < h; y++) {
        for (x = 0; x < w; x++) {
            canv[x][y] = painter(x, y);
        }
    }
}

function pix paint_mandelbrot(int x, int y) {
    return { red(x, y), green(x, y), blue(x, y), 0 }; 
}

graph(canvas, W, H, paint_mandelbrot);

draw(canvas, 0, 0);`
```
Test Suite

- Work through LRM
- Shell script to test all files in test suite (autotest.sh)

```c
int i, j, a[3][3];

function int foo(function int (int, int) fp, int x, int y) {
  x = x * 10;
  y = y + 1;
  return fp(x, y);
}

for (i = 0; i < a.size; i++){
  for (j = 0; j < a[i].size; j++){
    print(foo(function int (int x, int y) {
      return a[i][j] = 100 + x + y;}, i, j));
  }
}
```
The Process

● Weekly meetings initially
  ○ Turned into bi-weekly later into the process

● GitHub Repo
  ○ All forked from Oswin (he managed pull requests)

● Constant communication

● Prioritize certain tasks for deadlines
Takeaways

- Choose a project you are excited about
- Just because you hit a milestone doesn’t mean you’re done
- Create small goals
- Work together
- Understand the fundamentals of your language
QUESTIONS?