Motivation

- Create a general purpose programming language catered towards manipulating pixels in images
- Faster, easier vector/matrix manipulation
- More portable than image enhancing software
Basics

Vector:

int[5] a;
a = [1, 2, 3, 4, 5];

int[5] b;
b = [1, 2, 3, 4, 5]

Operations:

Operations:

Multiplication:

:) Dot product

:) Scalar

Addition

Subtraction
Basics

Matrix:
```
int[2][2] a;
a = [[1, 2] & [1,2]];
```

Operations:
- Multiplication
- Scalar
- Cross product
- Addition
- Subtraction
What else did we do?

- Matrix/vector operations
  - Inverse, Transpose
- Built-in Library

- Bitwise Operators
- Implicit Typecasting
- Explicit Typecasting
- Smiley comments! :)

- Implicit Typecasting
Architecture

source code → scanner.mll → parser.mly → ast.ml → semant.ml

stdlib.px

executable → codegen.ml ← sast.ml
Testing

- .px file extension
- Unit tests
- Test Cases for Pass/Fail
Lessons learned & contributions

Anthony: Scanner, Parser, AST, SAST, Semant, Codegen, Stdlib debug, compiler frontend
Start early and chunk the work so it’s more manageable

Teresa: Scanner, parser, AST, bitwise/float/Matrix literals in codegen, basic Matrix ops in stdlib
Have goals during your meetings

Gabriel – Hello World, Test Suite, semant, codegen, stdlib linking
Don’t just start early, do the bulk of the project early so you can add cool features later.

Brian: Scanner, parser, ast, sast, semant, codegen, stdlib
Put in consistent work; you’re gonna have to change the structure of your compiler a lot!
Milestones

- **October 2017**: Finished grammar
- **November 2017**: Hello World! Printed
- **December 2017**: Matrix manipulation
- **Demo Day**
What people are saying

I can automate my linear algebra homework now!

Gabriel Kramer-Garcia, NY

This is bad for my students!

Linear Algebra Prof, NYC
Demo