

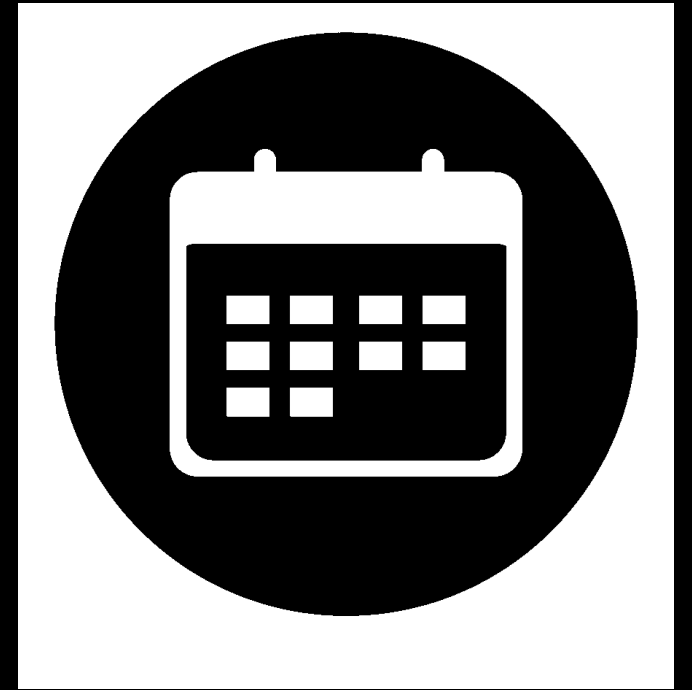


# GRAIL: A GEMSTONE PYTHON COMPILER

Adam Taylor

# ORIGINAL TIMELINE

- 4-5 weeks for Variable system
  - Meeting with mentor
  - 2 weeks refactoring variables
- Complete 5-6 built-in functions a week
- Bi-weekly meetings with my Mentor
- Stretch goal: implement the ability to import python c libraries



# UPDATED TIMELINE

- 5 weeks for Variable system
- 1 week refactoring variables
- 5 weeks completing 4-5 function a week
- 4 weeks on string injection
- 1 week implementing Boolean operators
- 4 weeks working on poster, write-up and presentation
- Bi-weekly meetings with my Mentor

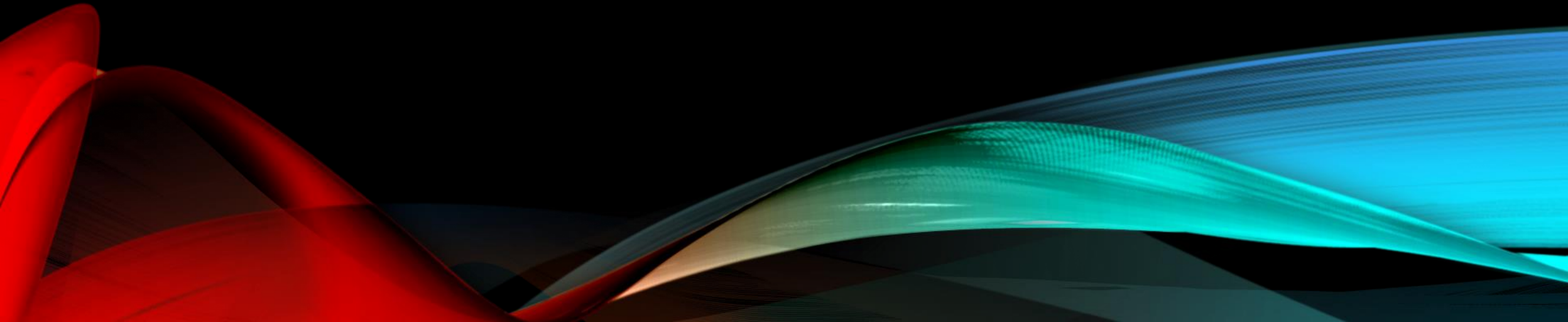


# VARIABLES AND SCOPING

- Built-ins
  - Created a class to serve as the base for all variable creation
  - A global Smalltalk variable holding a singleton of this type
  - Added Python built-in functions to this layer
  - Cannot be written to
- PyGlobals
  - Created on Python Program translation/execution
  - Persists throughout the translation execution
  - Allows variables to be created in it
  - Serves as the global layer for the global key word
- Variables
  - Serves as local variables
  - Created upon function call
  - Allows variables to be created
  - Can serve as a variable scope for the nonlocal keyword
- All scopes have read access to the variables in the parent scope.
- Done in 4 weeks

# VARIABLE METHODS

- global – From a Variables local scope run this method with a variable name to pair the local version with the variable of the same name in the global PyGlobals scope so they access the same value.
- nonlocal – Similar to the global method but works on the previous local scope instead of PyGlobals.



# PYTHON BUILT-IN FUNCTION RE-CREATION

Built-in Functions			
<b>A</b> abs() aiter() all() any() anext() ascii()	<b>E</b> enumerate() eval() exec()	<b>L</b> len() list() locals()	<b>R</b> range() repr() reversed() round()
<b>B</b> bin() bool() breakpoint() bytearray() bytes()	<b>F</b> filter() float() format() frozenset()	<b>M</b> map() max() memoryview() min()	<b>S</b> set() setattr() slice() sorted() staticmethod() str() sum() super()
<b>C</b> callable() chr() classmethod() compile() complex()	<b>G</b> getattr() globals()	<b>N</b> next()	<b>T</b> tuple() type()
<b>D</b> delattr() dict() dir() divmod()	<b>H</b> hasattr() hash() help() hex()	<b>O</b> object() oct() open() ord()	<b>V</b> vars()
	<b>I</b> id() input() int() isinstance() issubclass() iter()	<b>P</b> pow() print() property()	<b>Z</b> zip()
			<b>_</b> __import__()

## Class Methods

- `__repr__`
- `splice`
- `__str__`
- `__float__`
- `__bool__`
- `__update__`
- `__divmod__`
- `__add__`
- `__mult__`

## Functions Implemented

- `abs()`
- `len()`
- `type()` (base)
- `float()`
- `int()`
- `ord()`
- `char()`
- `pow()`
- `sum()`
- `round()`
- `locals()`
- `bool()`
- `repr()`
- `str()`
- `list()`
- `set()`
- `frozenset()`
- `dict()`
- `complex()`
- `input()`
- `globals()`

Average of 5 items implemented a week

# ITEMS COMPLETED NOT ON THE SCHEDULE

- Internal Python Execution – allows Python code that has been entered into a string to be translated and executed in the Smalltalk workspace.
- Fixed functions so that they properly take default arguments and respect required parameters.
- Replaced = with == for faster comparison where appropriate.
- Updated comments to more accurately fit functions and their Smalltalk parameters.
- Updated some existing methods to return the new Python version of an object instead of the Smalltalk one.
- Changed all mathematical methods to use double dispatch
- Implemented a string injections system for string formatting and type conversion.
- Implemented all Boolean operators and their parsers.



# ERRORS

- Name errors for non-existent variables and unauthorized writes
- Syntax errors for bad use of global and nonlocal keywords
- Dividing by zero error
- Type errors for improper function arguments
- Value errors for function given the wrong type or format of arguments

# TESTS

- Test suite for variables and associated key words
- Tests for each new built-in, ensuring both failures and success are tested.
- Updated existing tests for their new implementation