



Worm-lang Interpreter



Jake Armendariz



Initial Goal

Statically typed

Add functions, structs, arrays, string manipulation, hashmap

Useful

Easy to write neat code

Accomplished

Type System: weak: `map[any] = any`

Static Analysis: Types, Values, Scope

Features: Functions, Arrays, Hashmap, Errors Messages, Builtin Functions

Sample Programs

Types

```
pub enum VarType {
    Int,
    Float,
    Char,
    String,
    Array(Box<VarType>),
    Map,
}

pub enum Constant {
    Int(i32),
    Float(f64),
    String(String),
    Char(char),
    Array(VarType, Vec<Constant>), // Arrays are fixed size in worm, but it's easiest to implement with vec
    Index(String, Box<Expr>), // string for variable name, expr will be the key or index (array or hashmap)
    Map(WormMap), // custom type for hashmap
}
```

Static Analysis & Error Messages

Error messages checking for type errors, missing names, functions.

Challenges: Giving enough information on error message to help debug

```
[jake@macbook:~/Desktop/rworm]$ ./worm-pl worm/error.c
STATIC ERROR: in function 'scope_test' on line 4, error: variable 'd' does not exist
STATIC ERROR: in function 'provide_arr' on line 2, error: expected type 'string' recieved type 'char'
STATIC ERROR: in function 'sum_up' on line 3, error: function 'providearr' does not exist
STATIC ERROR: in function 'sum_up' on line 6, error: expected type 'int' recieved type 'string'
STATIC ERROR: static analysis caught 4 errors
```

Mistakes and Challenges

- Parser for error messages
- Cloning state/ast & lacking internal mutability
- Optimization
- Didn't add structs



Arith

Reimplemented hw1 arith in Worm

Time for a quick demo?



Thank you