Shoo

Claire Adams, Cindy Le, Sam Jayasinghe, Crystal Ren
About the Language
Language Overview

Shoo is a general-purpose programming language that is statically scoped and strongly typed. It has imperative and functional programming features with C-like syntax. Supporting first class functions, structs, and arrays, it can perform reasonably complex tasks in a single-threaded setting.
# Language Evolution

<table>
<thead>
<tr>
<th>Iteration 0</th>
<th>Iteration 1</th>
<th>Current iteration</th>
</tr>
</thead>
</table>
| ● Inspired by the Go language  
● Vision: concurrent programming, parallelizable problem solving, Go-routines  
● Channels, locks, threads, and first class functions. | ● Discard concurrent programming, locks  
● Discussion of linking a C library for Go's channels  
● “Shoo-routines” | ● Focus on first class functions (discard “shoo-routine” name)  
● Discard channels  
● Implement structs and nested arrays |
Key Language Features

- First-class Functions
- Structs
- Arrays
First-Class Functions

- Functions are treated like variables
- Can be fields in structs or elements in arrays
- Can be arbitrarily nested
- Can have recursive function
Arrays And Structs

- Arrays can hold any type including arrays, user defined structs and functions
- Struct fields can have default values and can have any number and type of member fields
About the Compiler
Compiler Architecture
Code
Demo Sample Projects

Demonstration of interesting features in our languages such as first-class functions, structs, and arrays.
Demo 1

Choose your own adventure with structs, functions and arrays:

- Choose an array size (for the array of structs)
- Choose a struct type (BankAccount, Rectangle, Point, Student)
- The current array of these objects prints (initial values are randomly generated)
- Choose a function to apply to the array of structs (these functions are stored in an array as well)
- The array after the function application is printed
Demo 2: Bubble Sort

Array of Structs

<table>
<thead>
<tr>
<th>Index</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>106</td>
</tr>
<tr>
<td>3</td>
<td>101</td>
</tr>
<tr>
<td>8</td>
<td>104</td>
</tr>
<tr>
<td>4</td>
<td>107</td>
</tr>
<tr>
<td>5</td>
<td>100</td>
</tr>
</tbody>
</table>

struct Object { /* ... */ }

function compareData(Object a, Object b) bool { /* ... */ }

function compareIndex(Object a, Object b) bool { /* ... */ }

/* printIndex() and printData() definitions here */

function bubbleSort(array<Object> arr, int n, func(Object, Object; bool) compare) array<Object> { /* ... */ }

/* Some initializations here */

printIndex(objects, n); // prints: 1 3 8 6 9 7 0 2 4 5
printData(objects, n);  // prints 106 101 104 108 105 103 102 109 107 100
bubbleSort(objects, n, compareIndex);

printIndex(objects, n); // prints: 0 1 2 3 4 5 6 7 8 9
bubbleSort(objects, n, compareData);

printData(objects, n);  // prints: 100 101 102 103 104 105 106 107 108 109
Demo 3: Sudoku Solver

- Multi-dimensional arrays
- Operates on a default board
Bonus Demo

- Uses 2D arrays.
- Reads a string from stdin and then prints it in a fun ASCII format.
Wrap Up
Future Work

- Automatic garbage collection
- Mutually recursive structs and functions
- Type inference
Questions?
Sources

This presentation uses images and gifs from the following sources:

- Key Language Features slide: https://giphy.com/gifs/key-nPlwhYMeBkis0
- Demo Sample Projects slide: https://hackernoon.com/presenting-your-code-beautifully-fdbab9e6fb68
- First-Class Functions slide: https://www.designcrowd.com/design/16148133

Presentation template by SlidesCarnival.