GOLD

Game Oriented Language for DnD
Team members

Ezekiel Reyna
Manager

Aidan Rivera
System Architect

Timothy Chung
Language Guru

Dennis Guzman
Tester
Table of Contents

Introduction & Background
Project Timeline
Development Environment
Syntax & Usage Architecture
Testing
Demo
Inspiration

GOLD is a language inspired by Go and C and uses basic primitives to make Game development easier.

Simple functions and basic primitives
Expressive control flows
Explicit typing
Features

Safety
  ▪ Gold is statically typed

Familiarity:
  ▪ Similar syntactically to Go and C and can be picked up easily by developer of any level
Development Environment
### Syntax

<table>
<thead>
<tr>
<th>Operator</th>
<th>Built-in types</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ ]</td>
<td>Bool  // bool my_bool = true</td>
</tr>
<tr>
<td>-, @</td>
<td>Int   // int my_int = 100</td>
</tr>
<tr>
<td>!, -(negation)</td>
<td>Float // float num = 12.89</td>
</tr>
<tr>
<td>+, -(subtraction)</td>
<td>String // string name = “hello”</td>
</tr>
<tr>
<td>*, /, %</td>
<td>Array // int[3] b = {0,0,1};</td>
</tr>
<tr>
<td>&lt;, &gt;, &lt;=, =&gt;</td>
<td>Pointer // string test = &quot;Hey&quot;;</td>
</tr>
<tr>
<td>==, !=</td>
<td>// string ~ptr = @test;</td>
</tr>
<tr>
<td>&amp;&amp;,</td>
<td></td>
</tr>
</tbody>
</table>
Syntax

### Function declaration

```plaintext
func sayHello(string name) void {
    string helloStr = sprintf("Hello %s", name);
    println(helloStr);
    return;
}

func main() int {
    sayHello();
    return 0;
}
```

### Control flow

```plaintext
func main() int {
    int x = 5;
    if (x > 2) {
        println("foo\n");
    } else {
        println("bar\n");
    }
    return 0;
}
```
Syntax

```plaintext
<Built-in functions>
print    // print string
println  // print with newline
sprint   // format and return a string
input    // takes an input
atoi     // string to int
rand     // random number
srand    // seed random number generator
```
Architecture

Scanner

Parser

Semantic Checker

Codegen

LLVM IR

Executable
Test suite

./run.sh -a
To easily run unit test
./run.sh func-simple-args
To run a single test program
./run.sh -l
To show all available test/example programs
Demo