The Go Programming Language: Notes

Moritz M. Konarski

September 25, 2020

Contents

Preface **1**

Tutorial **2**
Hello, World 2
Command Line Arguments 3

Preface

- created in 2007 at Google, unveiled in 2009
- supposed to be expressive, efficient, effective
- similarity to C, just updated
- leaves out features that make stuff too complex
- has automatic memory management
- suited for networked servers and tools for programmers, but general purpose
- Go tends to work better than untyped scripting languages – a nice feature
- open source and generally portable between every single platform
- ALGOL, C, Pascal → Go
- it gets is syntax from C, concurrency from CSP
- agenda of radical simplicity – a type system that is complex enough to be safer, but not too complex to be a burden
- code has very little hidden memory allocations
- lightweight threads (*goroutines*) are cheap to create, but practical for even large numbers

Tutorial

- shows the basic components of Go
- gives an overview of the features of Go

Hello, World

```go
package main

import "fmt"

func main() {
    fmt.Println("Hello, World")
}
```

- run a program using

```
go run hello_world.go
```

- to compile it and keep the executable

```
go build hello_world.go
```

- code is organized into packages, kinda like libraries or modules
- each package begins with a package declaration – states which package the file belongs to
- the Go standard library has over 100 packages for string manipulation, formatting (like `fmt`)
- `main` is special because it defines a standalone executable, not a labrary
- `func main()` is also the entry point of the program
- `import` tells the compiler which packages are needed for the program
- you must only import packages that are needed, any missing or unnecessary ones will prevent compilation
- `import` must follow `package`, after that it's fair game for all other stuff (`func`, `var`, `const`, `type`)
- new lines matter in Go code, they are basically converted to semicolons – which you can make but don't have to
- Go is very particular about formatting, so use `gofmt` or `go fmt` to format files correctly

Command Line Arguments