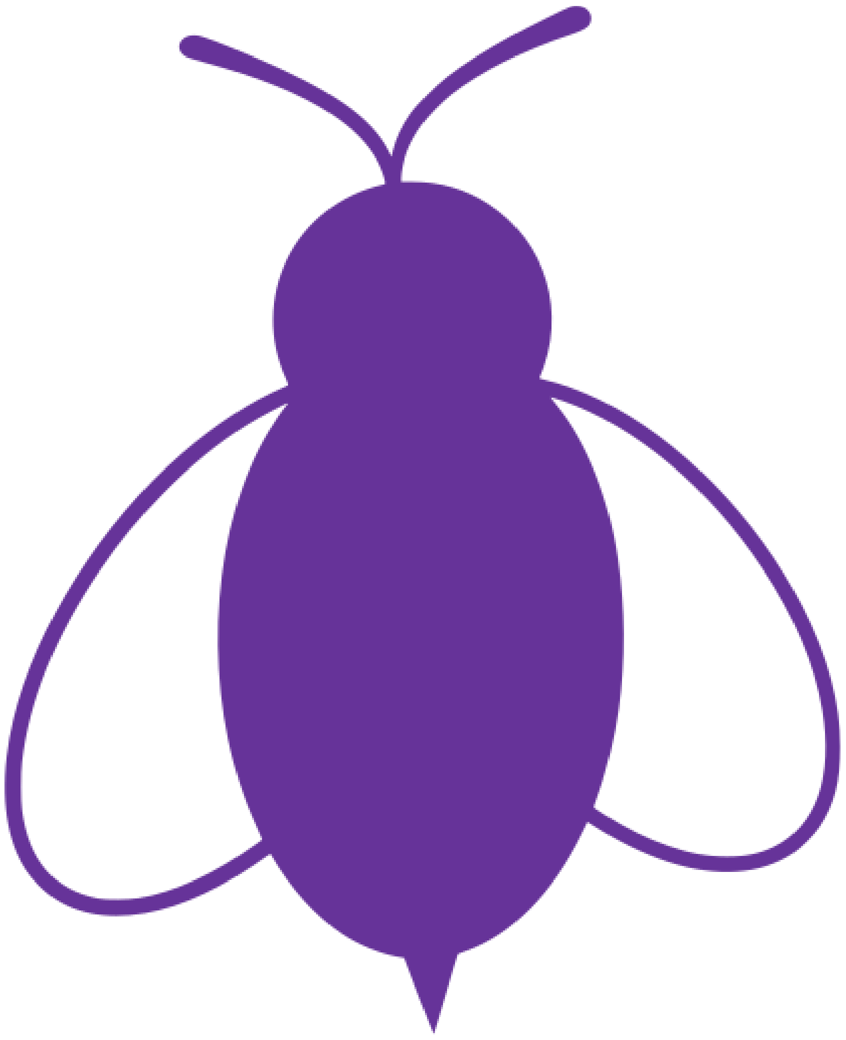
Buzzer

Find out what everyone's buzzing about!

Taeber Rapczak
taeber@ufl.edu

COP5618 Concurrent Programming
Spring 2019



https://cise.ufl.edu/~trapczak/buzzer

#Objectives

Primary

Use Go’s goroutines and channels to handle concurrency issues learned from COP5618 Concurrent Programming. Specifically, how to:

- provide mutual exclusion and atomic updates to parts of code;
- utilize processor resources when code blocks;
- minimize hazards by communicating instead of sharing memory.

Secondary

Write a non-trivial project in Go and report on the aspects of the languages and tools that were helpful in developing the software.

#Conclusions

Goroutines are the agents.

Goroutines, Channels, and `select` provide an alternative to traditional shared memory concurrency concepts.

It is still easy to have concurrency issues/bugs when using channels.

The Go runtime and tools provides useful detection of data races and issues arising from non-determinate behavior.

Go’s build tools provide convenient tooling to development software.

Go language constructs and library calls are used by the runtime to suspend Goroutines as needed.