

✓ **Congratulations! You passed!**

Go to next item

Grade received **100%** To pass 75% or higher

## Module 4 Quiz

Latest Submission Grade **100%**

1. What line of code could be used to define a loop which iteratively reads from a channel **ch1**?

1 / 1 point

- ☐ for l <- ch1
- ☒ for l := range ch1
- ☐ for l, err <- range ch1
- ☐ for l := ch1

✓ **Correct**  
Correct!

2. What does the **select** keyword do?

1 / 1 point

- ☐ Executes a set of case statements.
- ☒ Allows a choice of channels to wait on.
- ☐ Chooses the greatest of a set of numbers.
- ☐ Chooses an element from a list based on a user-defined criterion.

✓ **Correct**  
Correct!

3. What is the meaning of the **default** clause inside a **select**?

1 / 1 point

- ☒ The default clause is executed if all case clauses are blocked.
- ☐ The default clause is executed before any case clause is executed.
- ☐ The default clause is executed after any case clause is executed.
- ☐ The default clause is executed only if a case clause is executed.

✓ **Correct**  
Correct!

4. Suppose that there are two goroutines, g1 and g2, which share a variable x. X is initialized to 0. The only instruction executed by g1 is **x = 4**. The only instruction executed by g2 is **x = x + 1**. What is a possible value for x after both goroutines are complete?

1 / 1 point

- I. 0
- II. 1
- III. 4
- IV. 5

- ... >
- ☐ I and II only.
  - ☐ II and III only.
  - ☐ I, II, and III but not IV.
  - ☒ II, III, IV, but not I.

✓ **Correct**  
Correct!

5. What is mutual exclusion?

1 / 1 point

- ☐ When a single goroutine can execute only one of two blocks of code.
- ☐ When a single goroutine cannot execute a block of code.
- ☒ When multiple goroutines cannot execute blocks of code concurrently.
- ☐ When a single goroutine is the only goroutine which ever accesses a variable.

✓ **Correct**  
Correct!

6. What is true about deadlock?

1 / 1 point

- I. It can always be detected by the Golang runtime
  - II. Its caused by a circular dependency chain between goroutines
  - III. It can be caused by waiting on channels
- ☐ I and II only.
  - ☒ II and III only.
  - ☐ I and III only.
  - ☐ I, II, and III.

✓ **Correct**  
Correct!

7. What is the method of the sync.Mutex type which must be called at the beginning of a critical region?

1 / 1 point

- ☒ Lock()
- ☐ Unlock()
- ☐ Take()
- ☐ Block()

✓ **Correct**  
Correct!