Simultaneous Assignment

Simultaneous assignment is useful in many cases to make related assignments in a single statement. Sometimes they are required, either because only a single statement is available (e.g. in an if statement) or because the values will change after the statement (e.g. in the case of swap). All values on the right-hand side of the assignment operator are evaluated before the assignment is performed.

Simultaneous assignment in an if statement can improve readability, especially in test functions:

```
if got, want := someFunction(...), currTest.Expected; got != want {
    t.Errorf("%d. someFunction(...) = %v, want %v", currIdx, got, want)
}
```

Swapping two values is also made simple using simultaneous assignment:

```
i, j = j, i
```

https://go.dev/ref/spec#Assignments