

The screenshot shows an IDE with a Go project named 'golang_demo'. The Explorer panel on the left shows the project structure, including a 'slice' directory with files like 'test_slice3.go'. The main editor displays the code for 'test_slice3.go':

```
1 package main
2
3 import "fmt"
4
5 func testSlice(s []int) {
6     s[0] = 100
7     s = append(s, 200)
8 }
9
10 func main() {
11     arr := [5]int{1, 2, 3}
12     testSlice(arr[:4]) // 将数组转换为切片: arr[:]
13     fmt.Println(arr)   // [100 2 3 0 0] - 只有前3个元素被修改
14 }
15
```

The terminal at the bottom shows the command 'go run .\slice\test_slice3.go' being executed. It displays several error messages:

```
# command-line-arguments
slice/test_slice3.go:12:12: cannot use slice (variable of type [5]int) as *[5]int value in argument to testSlice
PS D:\Code_project\Golang\golang_demo\test> go run .\slice\test_slice3.go
# command-line-arguments
slice/test_slice3.go:7:13: invalid append: argument must be a slice; have s (variable of type *[5]int)
PS D:\Code_project\Golang\golang_demo\test> go run .\slice\test_slice3.go
[100 2 3 0 0]
PS D:\Code_project\Golang\golang_demo\test> go run .\slice\test_slice3.go
[100 2 3 0 200]
PS D:\Code_project\Golang\golang_demo\test>
```

The status bar at the bottom indicates 'Screen Reader Optimized', 'Ln 12, Col 21', and 'Tab Size: 4'.