

1번째 문제, 2번째 문제 둘 다 구조체의 값을 함수를 통해 바꿔주려고 하는데 call by value (값에 의한 참조)를 사용하면 함수 안에서만 값이 바뀌고 바뀐 값이 사라진다.(값만 넘어가서 계산하고 끝) 이렇게 되지 않기 위해서는 call by reference(주소에 의한 참조)를 이용해서 주소를 넘겨서 구조체의 값의 주소를 넘겨 해당 주소의 값을 바꿔주면 함수가 끝난 뒤에 해당 주소를 부르기 때문에 바뀐 값이 정상적으로 불리는 것을 알 수 있다. 부를 때는 &를 붙혀주고 해당 함수의 타입에는 * 포인터기호를 붙혀준다.

```
struct1.go | X struct2.go | X
week9 > struct1.go
1 package main
2
3 import "fmt"
4
5 type car struct {
6     name string
7     topSpeed float64
8 }
9
10 func nitroBoost(c *car) {
11     c.topSpeed += 50
12 }
13
14 func main() {
15     var mustang car
16     mustang.name = "Mustang Cobra"
17     mustang.topSpeed = 225
18     nitroBoost(&mustang)
19     fmt.Println(mustang.name)
20     fmt.Println(mustang.topSpeed)
21 }
22

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\spa89\gowork\week9> go run .\struct1.go
Mustang Cobra
275
PS C:\Users\spa89\gowork\week9>

struct2.go | X
week9 > struct2.go
1 package main
2
3 import "fmt"
4
5 type part struct {
6     description string
7     count int
8 }
9
10 func doublePack(p *part) {
11     p.count *= 2
12 }
13
14 func main() {
15     var fuses part
16     fuses.description = "Fuses"
17     fuses.count = 5
18     doublePack(&fuses)
19     fmt.Println(fuses.description)
20     fmt.Println(fuses.count)
21 }
22

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\spa89\gowork\week9> go run .\struct2.go
Fuses
10
PS C:\Users\spa89\gowork\week9>
```