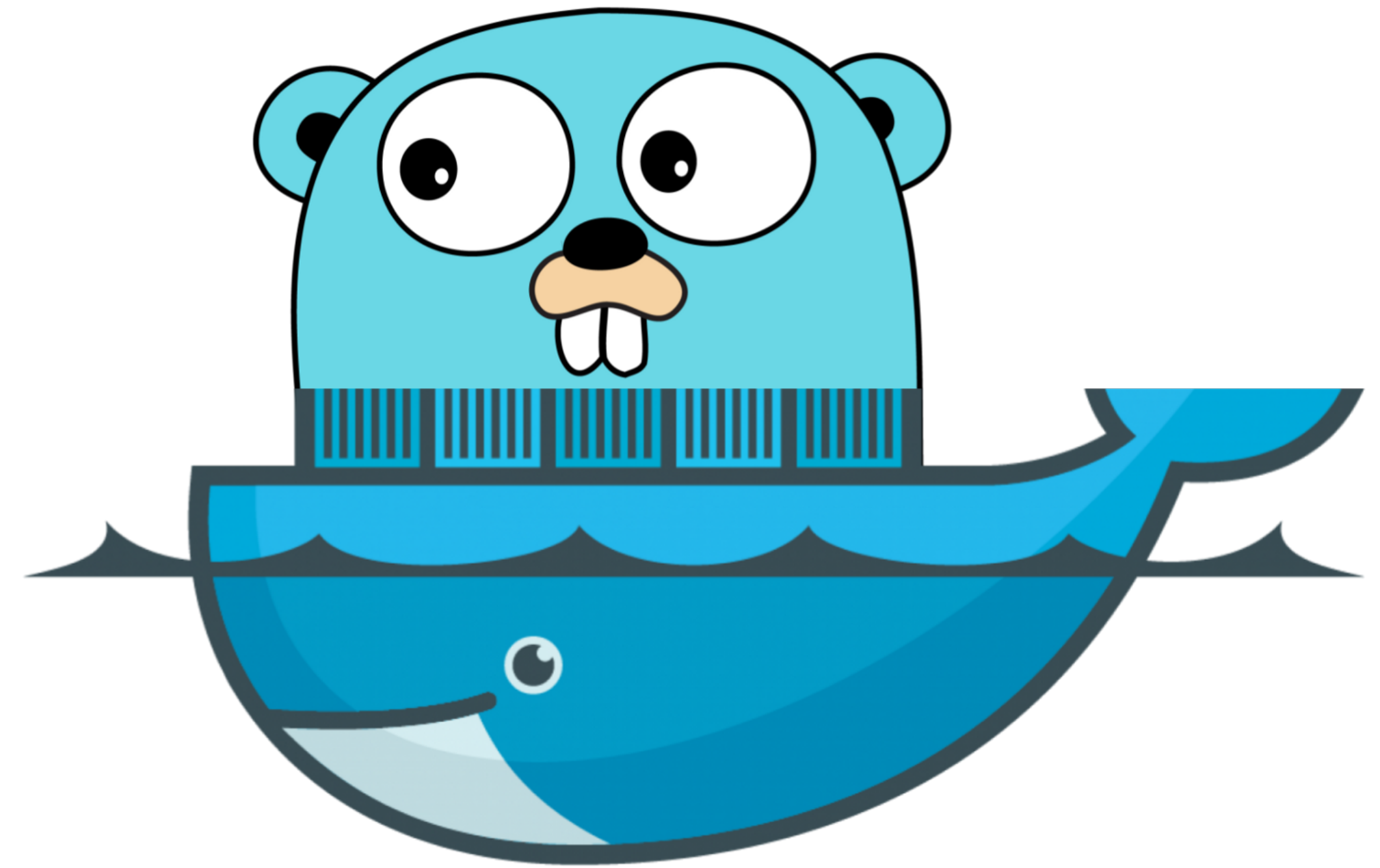


Pointer Explained

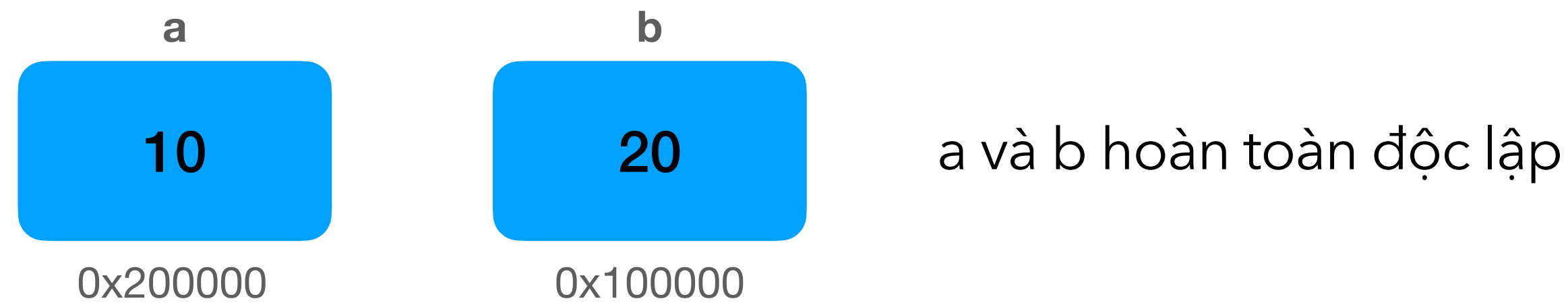


Pointer (con trỏ) thật ra tồn tại khắp nơi.
(Java, Js, Python, C#, PHP,...)

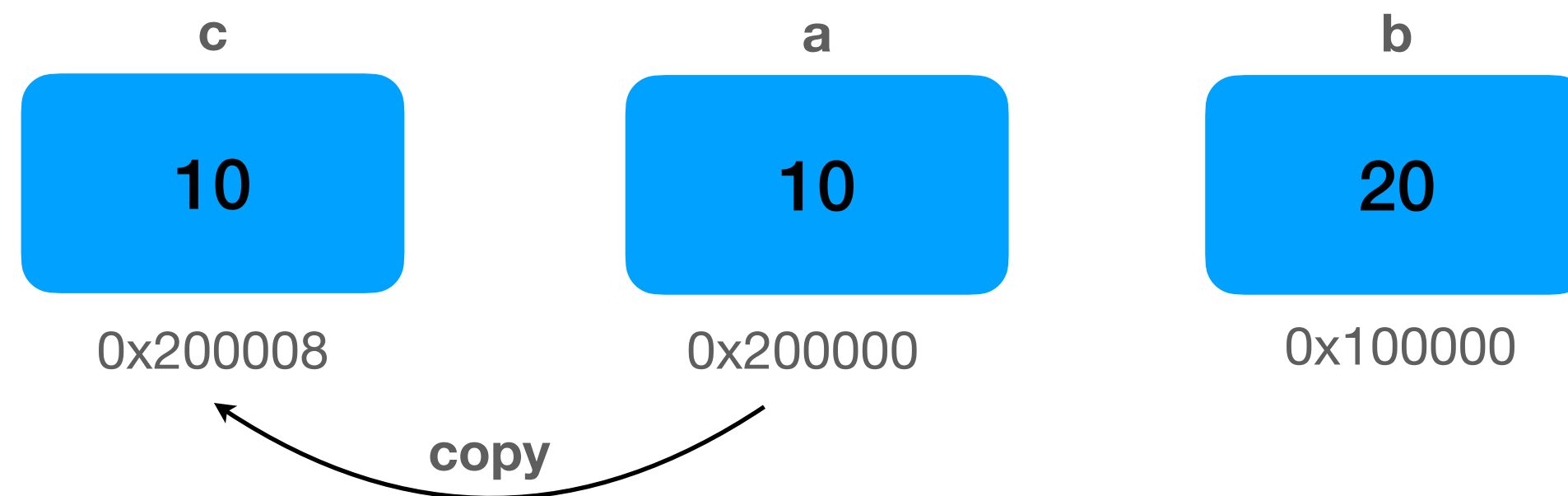
Pointer (con trỏ) trong Golang
là tương minh - Giống C

Value vs Pointer

Khi khai báo một biến: VD `a := 10; b := 20`



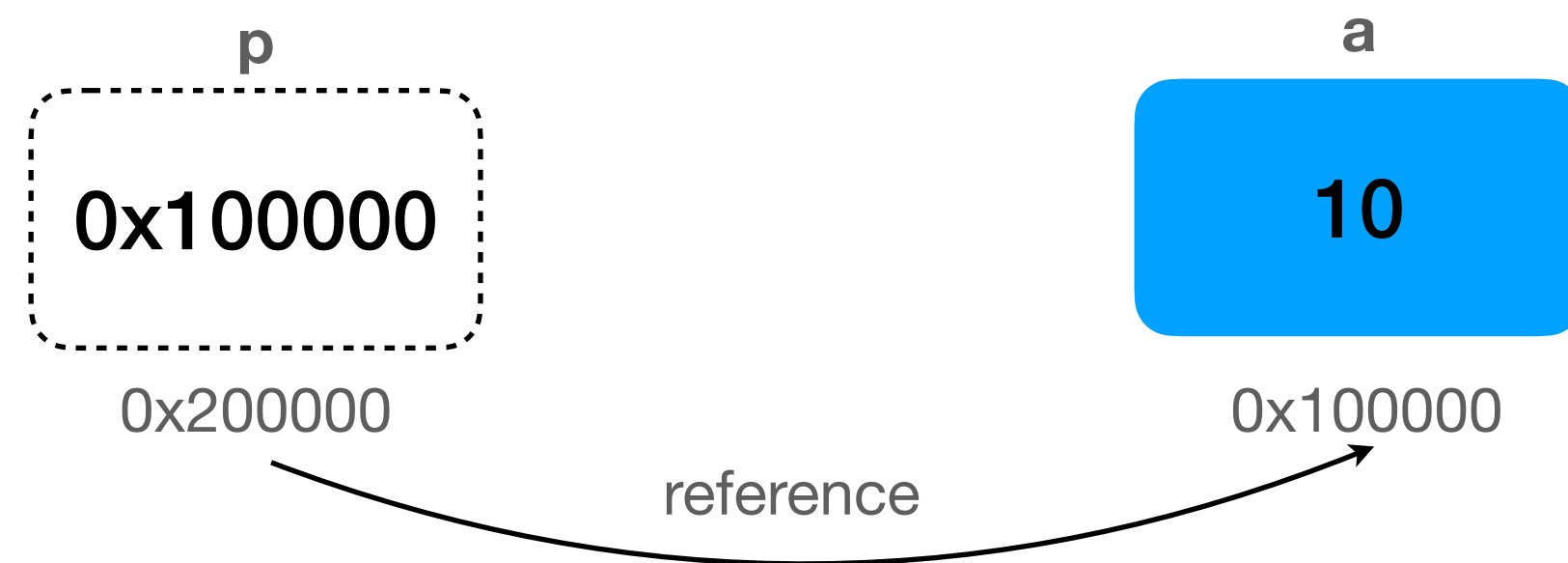
Giả sử khai báo thêm `c := a`



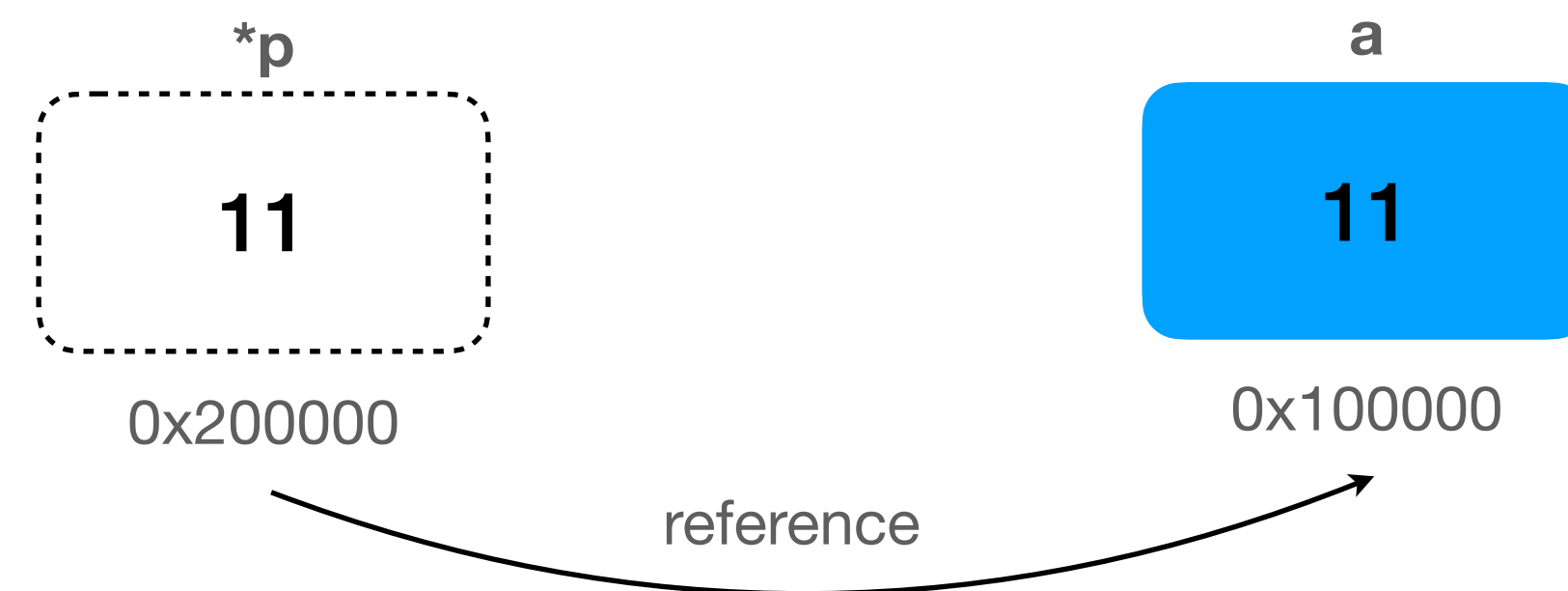
Từ đó "`c`" là **value**. Vì thực tế "`c`" là một biến độc lập, có vùng nhớ khác `a`, nhưng giá trị **copy** từ `a`.

Value vs Pointer (tt)

Khai báo pointer `p := &a`



Thay đổi giá trị a thông qua p: `*p = 11`



Từ đó "p" là **con trỏ - Pointer**. Vì thực tế "p" chỉ trỏ đến a. Bản thân "p" cũng có vùng nhớ của riêng nó, lưu trữ **address** của a.

```

1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     a := 10
9     p := &a
10
11     fmt.Println("addr of pointer p.      ", &p)
12     fmt.Println("addr of pointer p pointing to. ", p)
13     fmt.Println("addr of a.                  ", &a)
14     fmt.Println("value of pointer p.         ", *p)
15     fmt.Println("value of a.                  ", a)
16 }
17 // output will be:
18
19 // addr of pointer p.      0xc000102018
20 // addr of pointer p pointing to. 0xc000100010
21 // addr of a.              0xc000100010
22 // value of pointer p.     10
23 // value of a.             10

```

Value vs Pointer trong function

```
func fillCup(cup *Cup) {  
    cup.setFulfill(true)  
}
```

cup: con trỏ thuộc kiểu Cup (**tham chiếu**)

```
func fillCup(cup Cup) {  
    cup.setFulfill(true)  
}
```

cup: giá trị thuộc kiểu Cup (**tham trị**)

