



# 如何生成一个随机数

---



张华

64174234@qq.com

# 内容

---

1.1 生成随机数问题的价值

1.2 生成随机数的流程

1.3 生成随机数的Go代码

1.4 编译

1.5 总结

1.6 思考

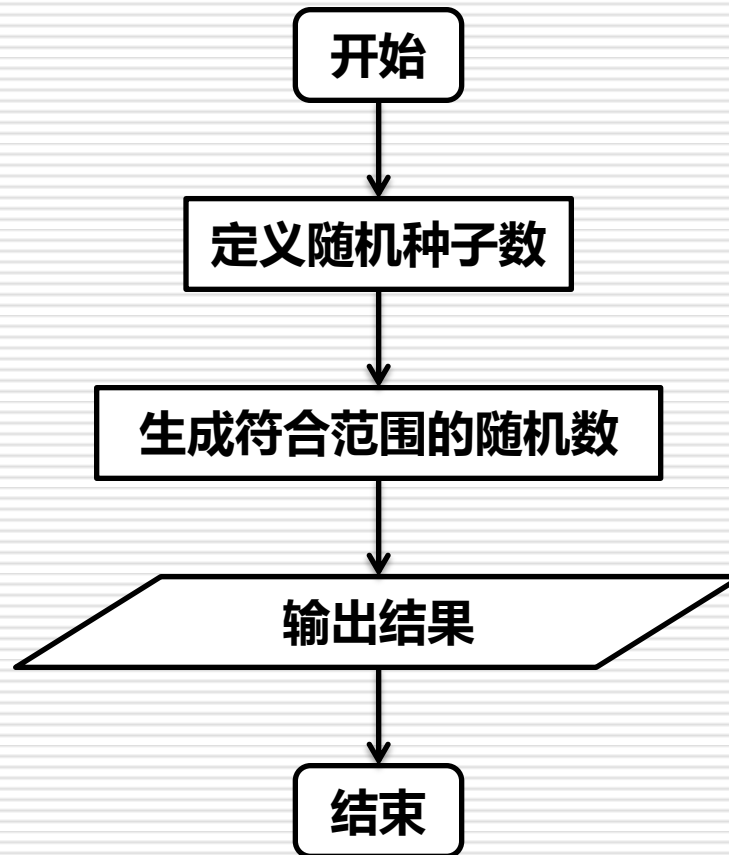
# 1.1 生成随机数问题的价值

---

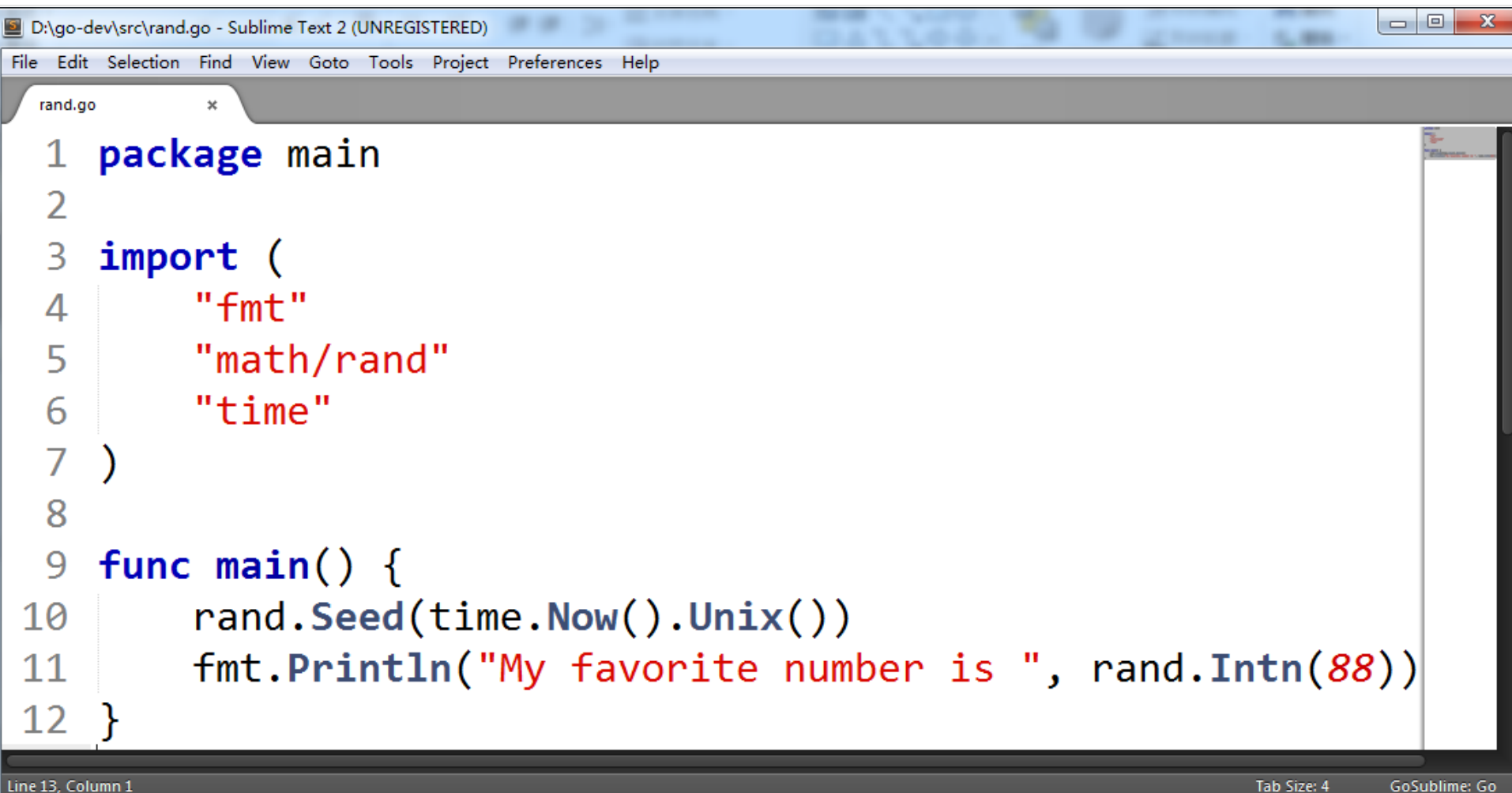
- 在编程的过程中，经常面临生成随机数的问题
  - 验证码的生成，用来防止恶意破解密码、刷票、论坛灌水、刷页。
  - 彩票随机出号
  - 随机点名系统
  - 各类模拟数据的随机样本生成
  - .....

## 1.2生成随机数的流程

---



# 1.3 生成随机数的Go代码

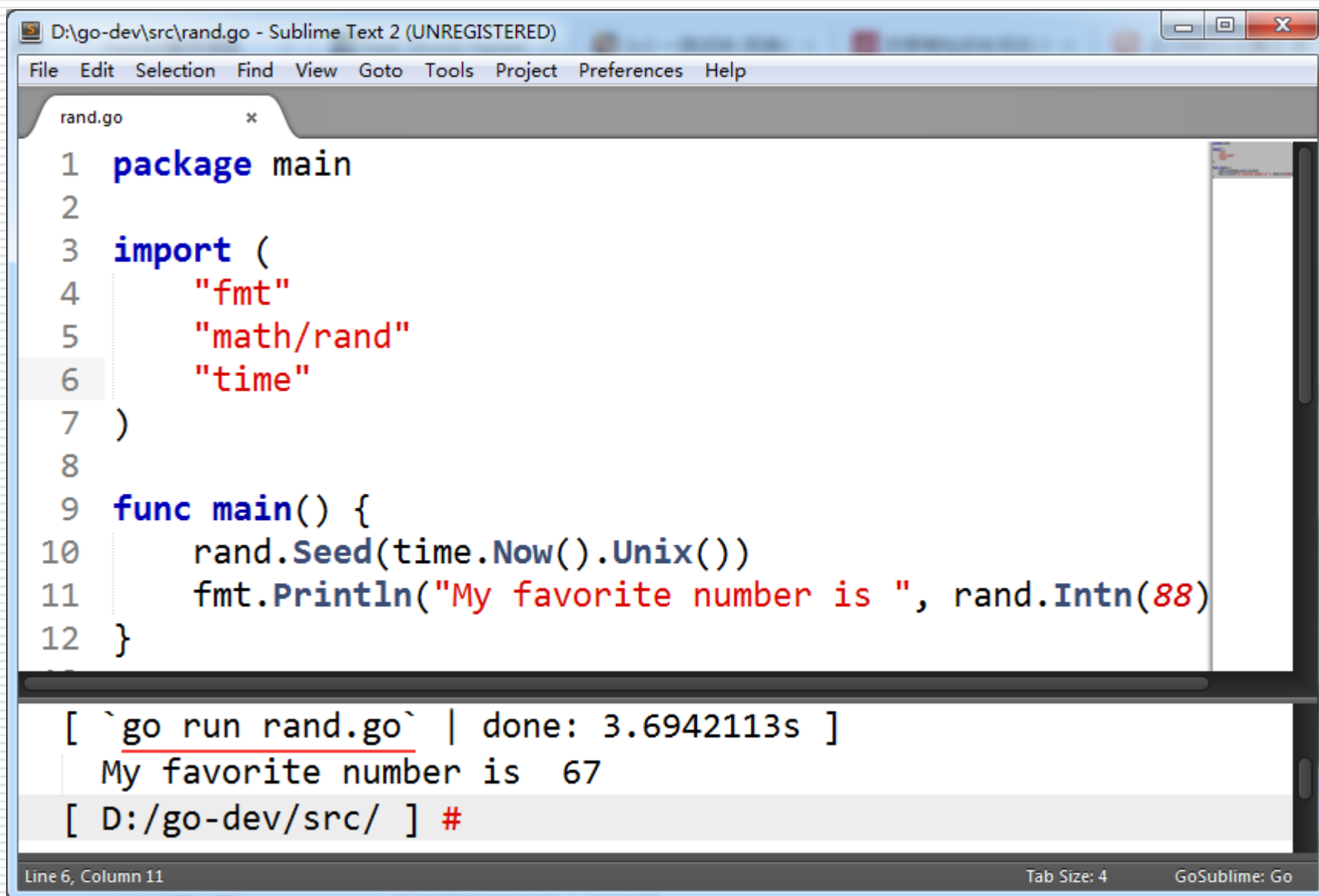


The image shows a screenshot of a Sublime Text 2 editor window. The title bar indicates the file path is D:\go-dev\src\rand.go and the editor is Sublime Text 2 (UNREGISTERED). The menu bar includes File, Edit, Selection, Find, View, Goto, Tools, Project, Preferences, and Help. The editor window has a tab labeled rand.go. The code is written in Go and is as follows:

```
1 package main
2
3 import (
4     "fmt"
5     "math/rand"
6     "time"
7 )
8
9 func main() {
10     rand.Seed(time.Now().Unix())
11     fmt.Println("My favorite number is ", rand.Intn(88))
12 }
```

The status bar at the bottom shows "Line 13, Column 1", "Tab Size: 4", and "GoSublime: Go".

# 1.4 编译



The screenshot shows a Sublime Text 2 window titled "D:\go-dev\src\rand.go - Sublime Text 2 (UNREGISTERED)". The editor contains a Go program in a file named "rand.go". The code defines a package "main", imports "fmt", "math/rand", and "time", and implements a "main" function that seeds a random number generator and prints a random number between 0 and 87. Below the editor, a terminal window shows the command "go run rand.go" being executed, resulting in the output "My favorite number is 67". The terminal prompt is "D:/go-dev/src/ #".

```
D:\go-dev\src\rand.go - Sublime Text 2 (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

rand.go x
1 package main
2
3 import (
4     "fmt"
5     "math/rand"
6     "time"
7 )
8
9 func main() {
10     rand.Seed(time.Now().Unix())
11     fmt.Println("My favorite number is ", rand.Intn(88))
12 }

[ `go run rand.go` | done: 3.6942113s ]
My favorite number is 67
[ D:/go-dev/src/ ] #

Line 6, Column 11 Tab Size: 4 GoSublime: Go
```

# 1.5 总结

---

## □ package main

- 源文件头部以 “package 包名” 开头
- 每个程序都是由包组成的
- 程序运行的入口是main包的main函数
- 只有package名为main的包才可以包含main函数  
，一个可执行程序有且仅有一个main包

# 1.5 总结(续)

---

□ import (

“fmt”

“math/rand”

“time”

)

■ 通过import关键字来导入其他非main包

□ 导入方式1：import “包名”

□ 导入方式2：import(

“包1”

“包2”

“...”

)

■ 按照惯例，包名与导入路径的最后一个目录一致

□ 如“math/rand”包的实现代码由package rand 开始



# 1.5 总结(续)

---

□ func main() {

.....

}

## ■ main函数的无参数定义模式

□ func 函数定义关键字

□ main() 函数名为main，且参数为空

□ {...} 函数体，用大括号括起来的若干语句集合，

□ Go语言规定“{...}”作为语句块的分隔符号，“{”不能单独成行，必须与函数声明或控制表达式在同一行。

# 1.5 总结(续)

---

- ❑ `rand.Seed(time.Now().Unix())` 以当前Unix时间作为随机种子
  - `time.Now().Unix()`
    - ❑ `time.Now()` 获取当前时间
    - ❑ `time.Now().Unix()` 转换为Unix时间戳或Unix时间
    - ❑ Unix时间戳的定义为格林威治时间1970年01月01日00时00分00秒（北京时间1970年01月01日08时00分00秒）起至现在的总秒数
    - ❑ `time.Unix(time.Now().Unix(), 0).String()` 时间戳到具体时间的转化
    - ❑ `time.Now().UnixNano()` 带纳秒的时间戳
  - 每执行一次程序，当前时间都在变，即每一次都有不同的随机种子，因此每次生成的随机数也不一样

## 1.5 总结(续)

---

- `fmt.Println( "My favorite number is " ,  
rand.Intn(10))` 生成一个10以内的随机数，并输出。
  - `rand.Intn(10)` 生成10以内的随机整数
  - `fmt.Println` 向控制台输出，多个参数多个输出，各参数用逗号分隔，从左到右输出并在最后自动换行
- 这个问题中导入了三个包，分别是`fmt`，`math/rand`,`time`
  - 导入了一个包之后，就可以调用其允许调用的方法，包名小写，方法首字母大写。
  - 如果导入的包没有用到，系统会报错。

# 1.5 总结(续)

---

## □ 可见性规则

- go语言中，使用大小写来决定该常量、变量、类型、接口、结构或函数是否可以被外部包含调用。
- 根据约定，函数名首字母小写即为private，函数名首字母大写即为public。
  - 如fmt.Println(...),rand.Intn(...),time.Now()

## □ 语句不用分号结束，加上也可，实际上Go会自动插入分号的

- 除非需要在同一行中写入多个语句
  - import "fmt" ;import "math/rand"; import time"
- 没有分号及只需少量的逗号和圆括号，使得go语言的程序更容易阅读

## 1.6 思考

---

- ❑ 计算机生成随机数的原理是什么？请查阅相关文献并总结。
- ❑ 为什么要先定义随机种子？
- ❑ 如何生成一个5到100之间的随机整数？
- ❑ 如何利用math/rand包生成随机小数？请尝试编程实现。
- ❑ 执行下面代码，将math.pi改为math.Pi再试着执行，请说出为什么会出现不同的情况？

```
1 package main
2
3 import (
4     "fmt"
5     "math"
6 )
7
8 func main() {
9     fmt.Println(math.pi)
10 }
```

---

**Thank you very much**

*Any comments and suggestions  
are beyond welcome*