channel

```
package main
import "fmt"
func main() {
    var ch chan int
    var count int
    go func() {
        ch <- 1
    }()
    go func() {
        count++
        close(ch)
    }()
    <-ch
    fmt.Println(count)
}
A: 不能编译
B: 输出1
C: 输出0
D: panic
```

```
package main
import (
    "fmt"
    "runtime"
    "time"
)
func main() {
    var ch chan int
    go func() {
        ch = make(chan int, 1)
        ch <- 1
    }()
    go func(ch chan int) {
        time.Sleep(time.Second)
        <-ch
    }(ch)
    c := time.Tick(1 * time.Second)
    for range c {
        fmt.Printf("#goroutines: %d\n", runtime.NumGoroutine())
    }
}
A: 不能编译
B: 一段时间后总是输出#goroutines: 1
C: 一段时间后总是输出#goroutines: 2
D: panic
```

```
关于channel，下面语法正确的是（）（多选）
A：var ch chan int
B: ch := make(chan int)
C: <-ch
D: ch<-
```

简要说明无缓冲的channel和有缓冲channel的区别?