

Laboratorio 5 - Selezione multiaria

1 Qual è l'output?

Supponendo che l'utente inserisca da **standard input** il valore `10`, cosa dovrebbe produrre in output il seguente programma? E se invece inserisse `4` ?

```
package main

import "fmt"

func main() {

    var voto int

    fmt.Scan(&voto)

    switch voto {
    default:
        fmt.Println("Insufficiente!")
    case 10:
        fallthrough
    case 9:
        fmt.Println("Ottimo!")
    case 8:
        fmt.Println("Distinto!")
    case 7:
        fmt.Println("Buono!")
    case 6:
        fmt.Println("Sufficiente!")
    }
}
```

2 Qual è l'output?

Supponendo che l'utente inserisca da **standard input** il valore `9`, cosa dovrebbe produrre in output il seguente programma? E se invece inserisse `13` ?

```
package main

import "fmt"

func main() {

    var n int

    fmt.Scan(&n)

    var somma int

    for i:=0; i<=n; i++ {

        switch i%2 {
        case 0:
            fmt.Println(i, "pari!")
        case 1:
            continue
        }

        somma += i
    }
}
```

```
    fmt.Println("Somma =", somma)
}
```

3 Qual è l'output?

Supponendo che l'utente inserisca da **standard input** il valore `9`, cosa dovrebbe produrre in output il seguente programma? E se invece inserisse `13`?

```
package main

import "fmt"

func main() {

    var n int

    fmt.Scan(&n)

    var somma int

    for i:=0; i<=n; i++ {

        switch i%2 {
        case 0:
            fmt.Println(i, "pari!")
        case 1:
            break
        }

        somma += i
    }

    fmt.Println("Somma =", somma)
}
```

4 Qual è l'output?

Supponendo che l'utente inserisca da **standard input** il valore `3`, cosa dovrebbe produrre in output il seguente programma? E se invece inserisse `100`?

```
package main

import "fmt"

func main() {

    var n int

    fmt.Scan(&n)

    accumulatore := 1

    for i:=1; i<=n; i++ {

        switch {
        case i<5:
            accumulatore *= i
        case i<10:
            accumulatore += i
        }
    }

    fmt.Println("Accumulatore =", accumulatore)
}
```

```
}
```

5 Qual è l'output?

Qual è l'output del seguente programma?

```
package main

import (
    "fmt"
)

func main() {

    var a1, b1 float64 = 5, 2

    var c1 complex128

    c1 = complex(a1, b1)           // La funzione complex restituisce un valore
                                   // complex128 con parte reale
                                   // uguale a 5 e parte immaginaria uguale a 2i
    fmt.Println(c1)                // Questa istruzione stampa a video "(5+2i)"

    var a2, b2 float32 = 6, 3

    c2 := complex(a2, b2)          // La funzione complex restituisce un valore
                                   // complex64 con parte reale
                                   // uguale a 6 e parte immaginaria uguale a 3i
    fmt.Println(c2)                // Questa istruzione stampa a video "(6+3i)"

    var c3 complex64 = 7 + 3i      // Altro modo per inizializzare un variabile
                                   // di tipo complex64, valido
                                   // anche per inizializzare un variabile di
                                   // tipo complex128

    parteReale := real(c3)         // La funzione real(c3) restituisce il
                                   // valore float32 della parte reale di c3
    fmt.Println(parteReale)        // Questa istruzione stampa a video "7";
    parteImmaginaria := imag(c1)   // La funzione imag(c3) restituisce il
                                   // valore float64 della parte immaginaria di c1
    fmt.Println(parteImmaginaria)  // Questa istruzione stampa a video "2";

    fmt.Println(c1+complex128(c2)+complex128(c3)) // Questa istruzione
                                                    // stampa a video "(18+8i)"
}
```

6 Soluzione di equazioni di primo grado

Scrivere un programma che legga da **standard input** due numeri reali, `a` e `b`, che rappresentano i coefficienti di un'equazione di primo grado (con incognita `x`) espressa nella forma $ax + b = 0$. Il programma deve calcolare e stampare a video il valore della radice dell'equazione (il valore di `x` per cui l'uguaglianza risulta verificata).

Suggerimento: Il valore della radice è pari a $-b/a$.

Esempio d'esecuzione:

```
$ go run equazione.go
3 6
La radice è -2

$ go run equazione.go
```

```
4 2
La radice è -0.5

$ go run equazione.go
10 -4
La radice è 0.4

$ go run equazione.go
10 -10
La radice è 1
```

Cosa succede se `a=0` e `b≠0` ?

Cosa succede se `a=0` e `b=0` ?

7 Soluzione di equazioni di secondo grado

Scrivere un programma che legga da **standard input** tre numeri reali, `a`, `b` e `c`, che rappresentano i coefficienti di un'equazione di secondo grado (con incognita `x`) espressa nella forma $ax^2 + bx + c = 0$. Il programma deve calcolare e stampare a video il valore delle radici dell'equazione.

Suggerimento: Nel caso in cui l'equazione non ammetta soluzioni reali, potete memorizzare i valori delle radici in variabili di tipo `complex`.

Nota: Si utilizzi il costrutto `switch case`.

Esempio d'esecuzione:

```
$ go run equazioni.go
1 5 4
Due radici reali -1 -4

$ go run equazioni.go
1 0 4
Due radici complesse (0+2i) (-0-2i)

$ go run equazioni.go
1 4 4
Due radici reali coincidenti -2 -2
```