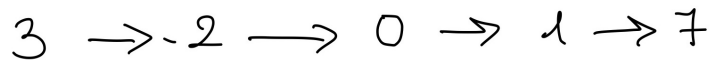


1 Esercizio filtro

- 1) Analizzate la seguente funzione ricorsiva, che ha per argomenti il puntatore p al primo nodo di una lista concatenata e un intero k :

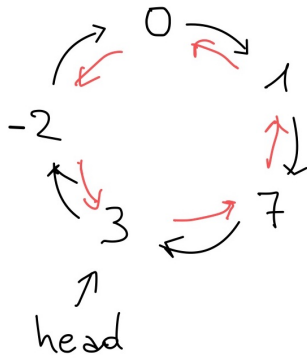
```
func f( p *node, k int ) int {  
    var a int  
    if p == nil {  
        return 0  
    }  
    a = 1 + f(p.next, k)  
    if a == k {  
        fmt.Println(p.val)  
    }  
    return a  
}
```

Rispondete alle seguenti domande, dove p punta al nodo con valore 3 della lista rappresentata qui sotto:



- Cosa stampa la chiamata $f(p, k)$ se k vale 1?
 - Cosa stampa la chiamata $f(p, k)$ se k vale 5?
 - Cosa stampa la chiamata $f(p, k)$ se k vale 10?
 - Completate la seguente frase: “Ricevendo il puntatore alla testa di una lista e un intero k la funzione f stampa... e restituisce...”. Includete nella descrizione anche i casi particolari/limite, se ve ne sono di rilevanti.
 - Qual è la complessità della funzione?
- 2) Definite un tipo strutturato `node` compatibile con la funzione `f` qui sopra.
- 3) Definite un nuovo tipo `circNode` che serva a rappresentare i nodi di una lista *circolare* contenente numeri interi. Una lista circolare è un tipo particolare di lista doppiamente concatenata, in cui l'ultimo nodo della lista e il primo sono collegati in modo da formare un circolo.

Esempio: la figura mostra una lista circolare il cui nodo iniziale è 3; il nodo finale 7 punta al nodo iniziale 3. In rosso sono indicati i puntatori verso il nodo precedente.

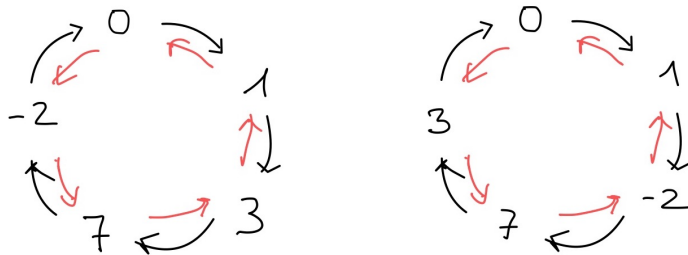


- 4) Scrivete una funzione `stampaDaZero(p *circNode)` che stampi una lista circolare partendo dal nodo di valore 0 e proseguendo circolarmente. Potete assumere che la lista circolare di cui fa parte `p` contenga esattamente un nodo con valore 0. Ad esempio, se `p` punta ad un nodo qualunque della lista circolare disegnata sopra, allora la funzione `stampaDaZero` deve stampare i numeri 0 1 7 3 -2.
- 5) Scrivete una funzione `main` che legge una serie di interi (uno per riga), crea una lista circolare contenente questi interi, e la stampi invocando la funzione `stampaDaZero`. Potete assumere che il vettore contenga esattamente uno zero.

Note per la consegna. Salvate il vostro programma con nome `es1-filtro.go`. Scrivete le vostre risposte in un commento all'interno dello stesso file.

2 Lista circolare

- 1) Considerate nuovamente la funzione f dell'esercizio 1. Senza usare la ricorsione, scrivete una funzione $f2$ che, ricevendo il puntatore al primo nodo di una lista circolare, produca lo stesso risultato della funzione f . Confrontate la complessità delle due versioni. Si potrebbe fare di meglio? Giustificate la risposta.
- 2) Scrivete una funzione `sposta(p *circNode)` che sposti il nodo puntato da p all'interno della lista circolare in base al suo valore: se è positivo, sposta il nodo in avanti di tante posizioni quanto indica il suo valore; se è negativo, sposta il nodo indietro di tante posizioni quanto indica il suo valore. Ad esempio, si consideri di nuovo la lista circolare disegnata per l'esercizio 1: spostando il nodo con valore 3 si ottiene la lista disegnata qui sotto a sinistra; spostando il nodo con valore -2 si ottiene la lista disegnata qui sotto a destra.



Note per la consegna. Scrivete le funzioni richieste in un file con nome `es2-circolare.go`. Scrivete le vostre risposte in un commento all'interno dello stesso file.