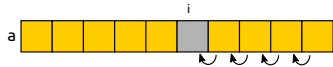


Go Slice Tricks Cheat Sheet

<https://github.com/golang/go/wiki/SliceTricks>

@ueokande

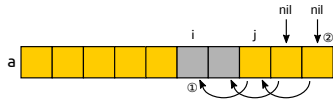
Delete



```
a = append(a[:i], a[i+1:]...)
```

```
a = a[:i+copy(a[i:], a[i+1:])] 
```

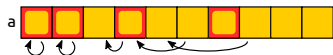
Cut (GC)



```
copy(a[i:], a[j:])  
for k, n := len(a)-j+i, len(a); k < n; k++  
    a[k] = nil // or the zero value of T  
}
```

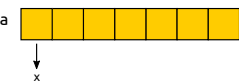
```
a = a[:len(a)-j+i]
```

Filter (in place)



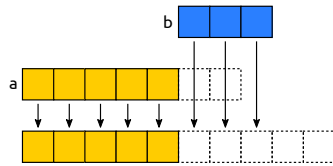
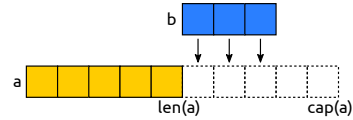
```
n := 0  
for _, x := range a {  
    if keep(x) {  
        a[n] = x  
        n++  
    }  
}  
a = a[:n]
```

Pop Front/Shift



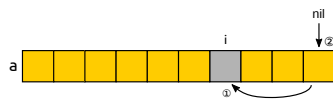
```
x, a = a[0], a[1:]
```

AppendVector



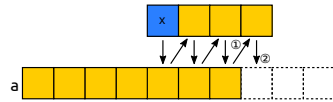
```
a = append(a, b...)
```

Delete without preserving order (GC)



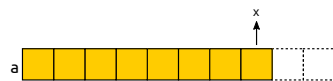
```
a[i] = a[len(a)-1]  
a[len(a)-1] = nil  
a = a[:len(a)-1]
```

Insert



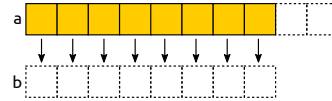
```
a = append(a[:i], append([]T{x}, a[i:]...).
```

Pop



```
x, a = a[len(a)-1], a[:len(a)-1]
```

Copy

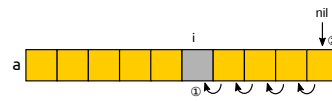


```
b = make([]T, len(a))  
copy(b, a)
```

```
b = append([]T(nil), a...)
```

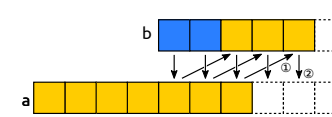
```
b = append(a[:0:0], a...)
```

Delete (GC)



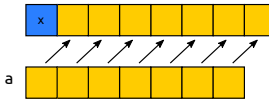
```
if i < len(a)-1 {  
    copy(a[i:], a[i+1:])  
}  
a[len(a)-1] = nil // or the zero value of T  
a = a[:len(a)-1]
```

InsertVector



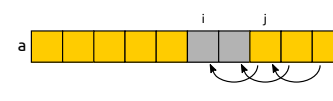
```
a = append(a[:i], append(b, a[i:]...))
```

Push Front/Unshift



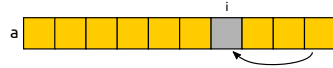
```
a = append([]T{x}, a...)
```

Cut



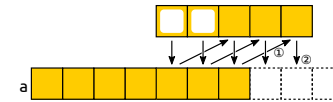
```
a = append(a[:i], a[j:]...)
```

Delete without preserving order



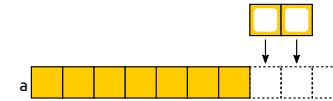
```
a[i] = a[len(a)-1]  
a = a[:len(a)-1]
```

Expand



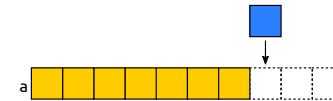
```
a = append(a[:i], append(make([]T, j), a[i:]
```

Extend



```
a = append(a, make([]T, j)...) 
```

Push



```
a = append(a, x)
```