



Bay! ИИ готовит к ЕГЭ по информатике

[Попробовать](#)

Урок Спринт 1.6

Длина и ёмкость слайса

У каждого слайса есть длина — то есть количество элементов, на которые ссылается слайс. Она не может выходить за рамки исходного массива. Обращаться к элементам слайса за пределами его длины нельзя, даже если в массиве есть и другие элементы.

Рассмотрим вот такой код:

```
a := []int{0, 1, 1, 2, 3}
b := a[:3]

b[4] = 42 // panic
```

Почему код сработал так? Потому что у слайса `b` длина равна 3. Несмотря на то, что в исходном массиве есть четвёртый элемент, адресоваться к нему через слайс `b` нельзя. При этом `b` и `a` ссылаются на один и тот же массив, а значит имеют одинаковую ёмкость.

Длину слайса можно проверить через встроенную функцию `len`, ёмкость — функцией `cap`:

```
a := []int{0, 1, 1, 2, 3}
fmt.Println(len(a), cap(a)) // 5, 5

b := a[:3] // [0, 1, 1]
fmt.Println(len(b), cap(b)) // 3, 5
```

Знание ёмкости слайса нам пригодится позднее.

Append

Длина слайса — его неизменяемое свойство, но функция `append` всё же может её «увеличить», то есть вернуть новый слайс с новой длиной.

Рассмотрим вот такой код:

```
a := []int{0, 1, 1, 2, 3}
```

```
a = append(a, 5)

b = append(a, 8)

fmt.Println(a)
fmt.Println(b)
```

► Что будет выведено на экран и почему?

Первый вызов `append` возвращает новый слайс, который присваивается старой переменной. (Старый слайс будет удалён сборщиком мусора.) Второй вызов `append` возвращает новый слайс, и он присваивается новой переменной. Исходный слайс `a` при этом не изменяется.

Справка

Исключительное право на учебную программу и все сопутствующие ей учебные материалы, доступные в рамках сервиса, принадлежат АНО ДПО «Образовательные технологии Яндекса». Воспроизведение, копирование, распространение и иное использование программы и материалов допустимо только с предварительного письменного согласия АНО ДПО «Образовательные технологии Яндекса».

[Пользовательское соглашение](#).

© 2018 – 2024 ООО «Яндекс»