

Урок 6. Сложные типы данных: указатели

1. Проанализируйте задания предыдущих уроков.

а. В каких случаях необходима была явная передача указателя в качестве входных параметров и возвращаемых результатов или в качестве приёмника в методах?

1 например нам надо чтоб изменения в из функции прошли локально

2 для экономии памяти например если у нас [оч.большая](#) структура то не самая хорошая идея копировать эти данные

б. В каких случаях мы фактически имеем дело с указателями при передаче параметров, хотя явно их не указываем?

Фактически имеем дела с указателем когда создаем slice

Слайс в Go — это структура, где одно из полей это указатель на массив: `type slice struct { array unsafe.Pointer len int cap int }`.

2. Для арифметического умножения и разыменования указателей в Go используется один и тот же символ — оператор (*). Как вы думаете, как компилятор Go понимает, в каких случаях в выражении имеется в виду умножение, а в каких — разыменование указателя?

1 При умножении у нас всегда два операнда, а при разыменовании указателя - один.

2 При выполнении каких то математических операции ,* всегда должно стоять перед указателем. Например `a + *b`.

3 Между звездочкой и переменной не может быть пустого места