Текстовый вариант видеоурока из предыдущего шага

Помимо такого чтения канала, мы ещё можем читать канал в цикле. Здесь мы можем использовать уже знакомое нами ключевое слово range, которое будет итерироваться по каналу и получать значения. В данном случае range читает из канала и в переменную i будет класть значение из канала… В данном случае range читает из канала значение и присваивает их в переменную i. И здесь стоит помнить о том, что мы будем блокироваться, пока канал не закроют. То есть цикл будет продолжать пытаться читать из канала и блокироваться. Как только канал закрывается – цикл прервется.

Как же это происходит? Здесь у нас уже есть функция writeChan, которая, наоборот, пишет в канал. Смотрите, здесь новым является передача канала в функцию. В Go можно указывать во время передачи в канал, какой это может быть канал: только на чтение или только на запись? Если мы хотим, чтобы в функции был доступ только на запись, мы ставим стрелочку справа. Если мы поставим стрелку слева – это означает, что канал внутри функции будет доступен только на чтение и как вы видите, здесь уже (НРЗ 01:30) ругается, что записать в канал нельзя, потому что он доступен только на чтение.

Давайте поправим, и внутри этой функции у нас в цикле от 1 до 5 в канал помещаются значения. После цикла мы как раз закрываем наш канал с помощью функции close, в которой и передаем наш канал. Здесь в функции main() мы создаем наш канал, как вы видите – это небуферизированный канал, и в горутине запускаем запись в этот канал. То есть что будет происходить в горутине? Она отдельно запустится, в цикле будет записывать в канал и после записи каждого значения заблокируется и будет ждать, пока из канала кто-то прочитает это значение. Как только значение будет прочитано – мы перейдем в следующую итерацию и запишем в канал следующее значение.

В функции же main мы итерируемся по нашему каналу и достаём оттуда значения, выводя их на экран. Давайте запустим и посмотрим на результат. Что и ожидалось: мы просто по очереди вывели значения от 1 до 5.

Что будет в случае, если мы не закроем наш канал? В этом кейсе мы, на самом деле, зависнем в цикле for, потому что range будет продолжать пытаться читать наш канал, а записывающих туда уже не будет, так как весь цикл for на запись пройдет и дальше горутина завершится. Давайте посмотрим, к чему приведет запуск программы. И как мы видим, мы вывели все 5 результатов записи в наш канал, после этого цикл for завис и заблокировался, и мы получили опять-таки deadlock!