

1. При заданном натуральном числе  $n$  реализуйте функцию, которая возвращает список, представляющий все числа от  $n$  до 1. Если  $n$  значение равно нулю, то должен быть возвращен пустой список.

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-jav
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
[]

Process finished with exit code 0
```

```
fun lol(n: Int): List<Int>{
    if (n < 1) return listOf()
    return (n downTo 1).toList()
}
fun main() {
    println(lol(10))
    println(lol(0))
}
```

2. Учитывая положительное целое число,  $n$  реализуйте функцию, которая возвращает список, представляющий форму generatePyramid с  $n$  уровнями. Функция должна консольно регистрировать форму generatePyramid с  $N$  уровнями, используя символ #. Убедитесь, что в generatePyramid есть пробелы как с левой, так и с правой стороны.

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.
#
###
#####
#####
#####
#####
#
###
#####
#####
#####
#####
#####
```

```
#####
#####
#####
#####
#####
#####
#####
#
Количество уровней должно быть положительным
Количество уровней должно быть положительным
```

```

fun generatePyramid(n: Int) {
    if (n <= 0) {
        println("Количество уровней должно быть положительным")
        return
    }
    val maxW = 2 * n - 1
    for (level in 1..n) {
        val numH = 2*level-1
        val hashStr = "#".repeat(numH)
        val padding = (maxW - numH)/2
        val levelStr = " ".repeat(padding) +hashStr
        println(levelStr)
    }
}
fun main() {
    generatePyramid(5)
    generatePyramid(10)
    generatePyramid(1)
    generatePyramid(0)
    generatePyramid(-1)
}

```

3. Написать программу, реализующую Шифр цезаря

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.
Оригинал: Hello, World! 123
Сдвиг 3: 'Khoor, Zruog! 123'
Сдвиг -1: 'Gdkkn, Vnqkc! 123'

Process finished with exit code 0

```

```

fun sorokdva(text: String, shift: Int): String {
    val effShift = shift % 26
    return text.map {char ->
        when (char) {
            in 'A'..'Z' -> {
                val shifted = 'A' + (char - 'A' + effShift + 26) % 26
                shifted
            }
            in 'a'..'z' -> {
                val shifted = 'a' + (char - 'a' + effShift + 26) % 26
                shifted
            }
            else -> char
        }
    }.joinToString("")
}
fun main() {
    val text = "Hello, World! 123"
    val shift = 3
    val negShift = -1
    println("Оригинал: $text")
    println("Сдвиг $shift: '${sorokdva(text, shift)}'")
    println("Сдвиг $negShift: '${sorokdva(text, negShift)}'")
}

```

4. При заданном натуральном числе n реализуем функцию, которая возвращает список чисел от 1 до n. Однако для чисел, кратных трем, список должен содержать слово Физллл вместо числа, а для чисел, кратных пяти, список должен содержать слово Бизлллл. Для чисел, кратных трем и пяти, список должен содержать ВизллБизлл.

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Com
[1, 2, Физллл, 4, Бизлллл, Физллл, 7, 8, Физллл, Бизлллл, 11, Физллл, 13, 14, ВизллБизлл]
[]

Process finished with exit code 0
```

```
fun blablalba(n: Int): List<String>{
    if (n < 1) return listOf()
    return (1..n).map { num ->
        when{
            num % 3 == 0 && num % 5 == 0 -> "ВизллБизлл"
            num % 3 == 0 -> "Физллл"
            num % 5 == 0 -> "Бизлллл"
            else -> num.toString()
        }
    }
}

fun main() {
    println(blablalba(15))
    println(blablalba(0))
}
```