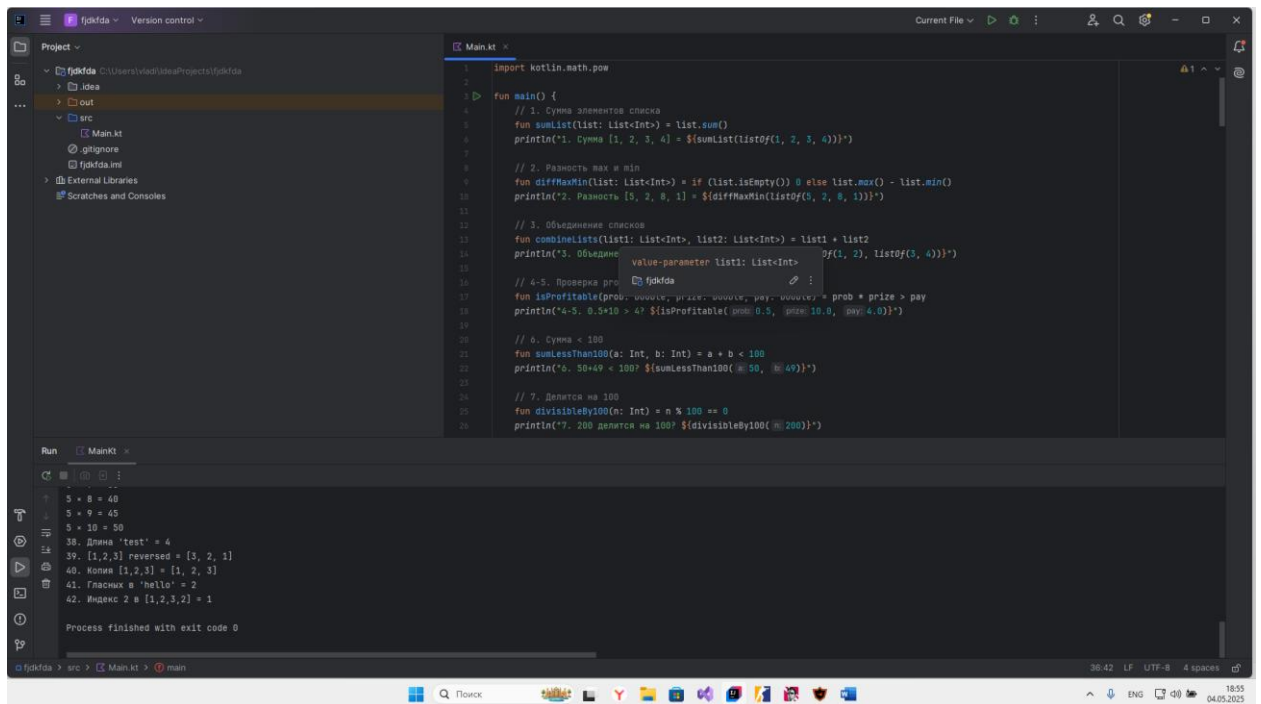


Лабораторная работа №8

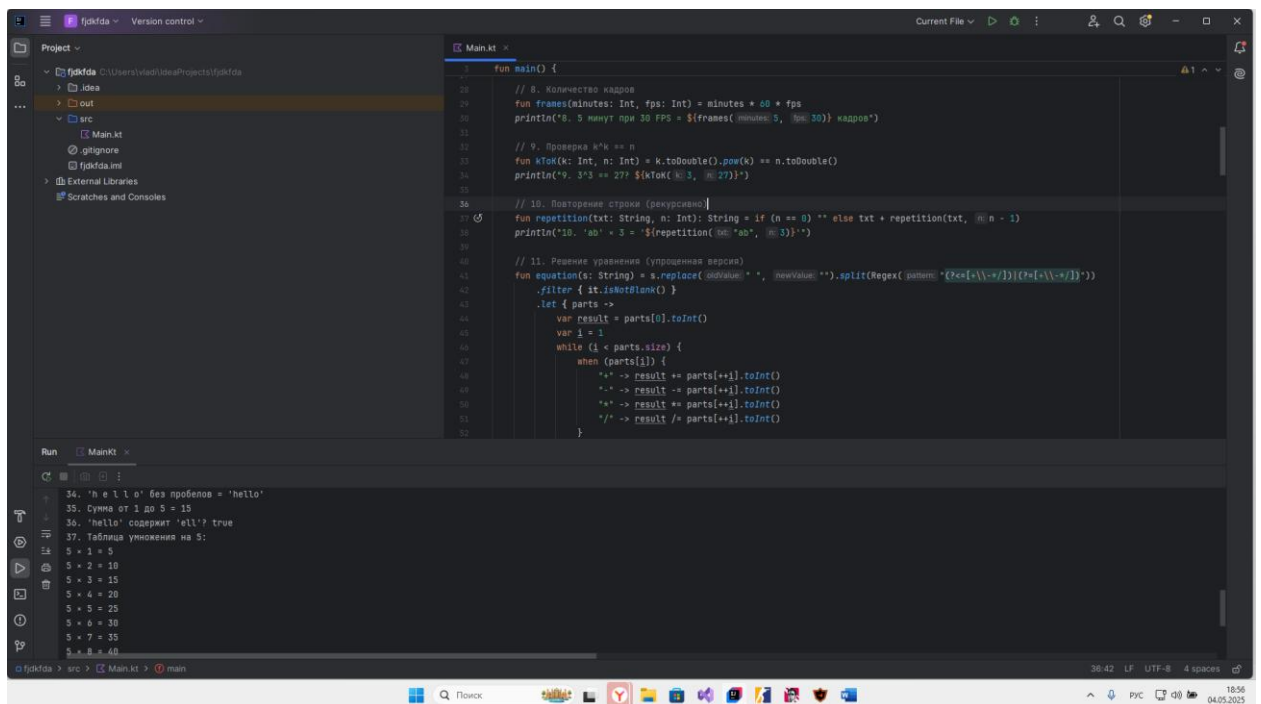
Харахардин В. ИС233



```
1 import kotlin.math.pow
2
3 fun main() {
4     // 1. Сумма элементов списка
5     fun sumList(list: List<Int>) = list.sum()
6     println("1. Сумма [1, 2, 3, 4] = ${sumList(listOf(1, 2, 3, 4))}")
7
8     // 2. Разность max и min
9     fun diffMaxMin(list: List<Int>) = if (list.isEmpty()) 0 else list.max() - list.min()
10    println("2. Разность [5, 2, 8, 1] = ${diffMaxMin(listOf(5, 2, 8, 1))}")
11
12    // 3. Объединение списков
13    fun combineLists(list1: List<Int>, list2: List<Int>) = list1 + list2
14    println("3. Объединение value-parameter list1: List<Int> 3f(1, 2), listOf(3, 4))")
15
16    // 4-5. Проверка про
17    fun isProfitable(prob: Double, prize: Double, pay: Double) = prob * prize > pay
18    println("4-5. 0.5*10 > 4? ${isProfitable(0.5, prize=10.0, pay=4.0)}")
19
20    // 6. Сумма < 100
21    fun sumLessThan100(a: Int, b: Int) = a + b < 100
22    println("6. 50+49 < 100? ${sumLessThan100(50, 49)}")
23
24    // 7. Делится на 100
25    fun divisibleBy100(n: Int) = n % 100 == 0
26    println("7. 200 делится на 100? ${divisibleBy100(200)}")
27
28    // 8. Количество кадров
29    fun frames(minutes: Int, fps: Int) = minutes * 60 * fps
30    println("8. 5 минут при 30 FPS = ${frames(minutes=5, fps=30)} кадров")
31
32    // 9. Проверка k*n == n
33    fun kTok(k: Int, n: Int) = k.toDouble().pow(k) == n.toDouble()
34    println("9. 3^3 == 27? ${kTok(3, 27)}")
35
36    // 10. Повторение строки (рекурсивно)
37    fun repetition(txt: String, n: Int): String = if (n == 0) "" else txt + repetition(txt, n - 1)
38    println("10. 'ab' * 3 = ${repetition("ab", 3)}")
39
40    // 11. Решение уравнения (упрощенная версия)
41    fun equation(s: String) = s.replace(Regex("ab", "newvalue"), "newvalue").split(Regex("(?<=[+\\-*/])(?=[+\\-*/])"))
42    .filter { it.isNotBlank() }
43    .let { parts ->
44        var result = parts[0].toInt()
45        var i = 1
46        while (i < parts.size) {
47            when (parts[i]) {
48                "+" -> result += parts[i+1].toInt()
49                "-" -> result -= parts[i+1].toInt()
50                "*" -> result *= parts[i+1].toInt()
51                "/" -> result /= parts[i+1].toInt()
52            }
53            i++
54        }
55    }
56}
```

Run Main.kt

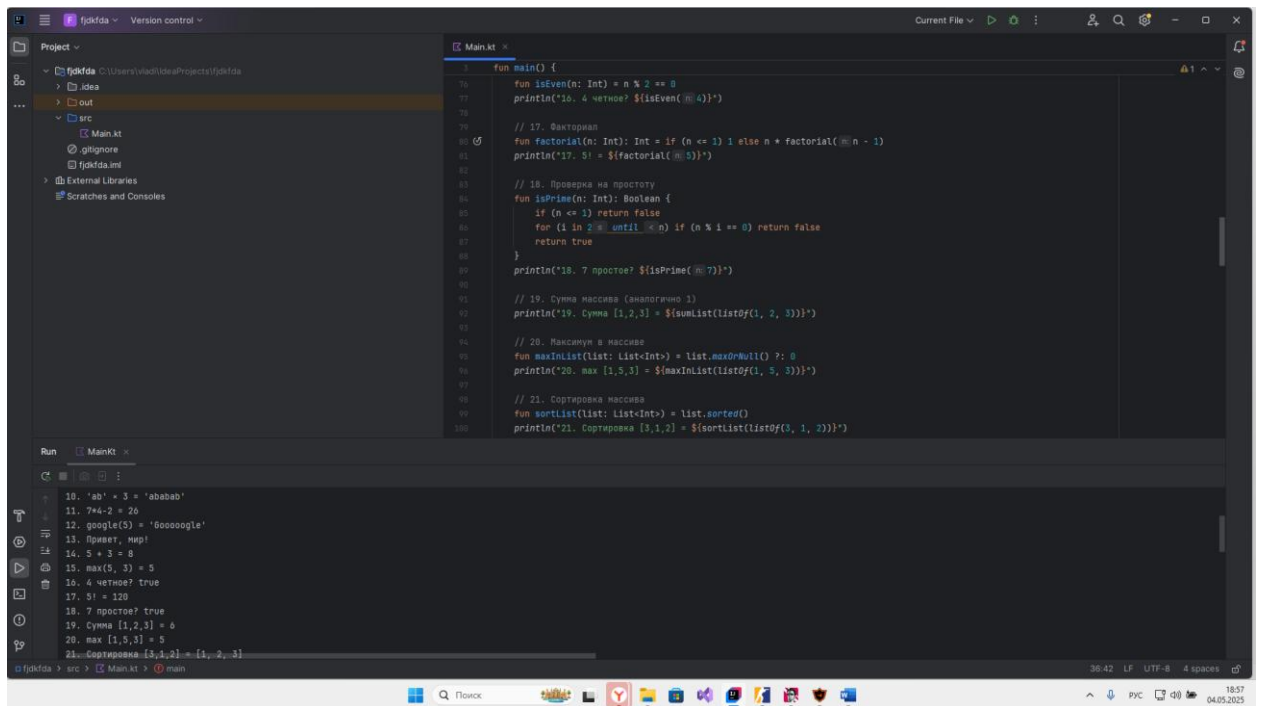
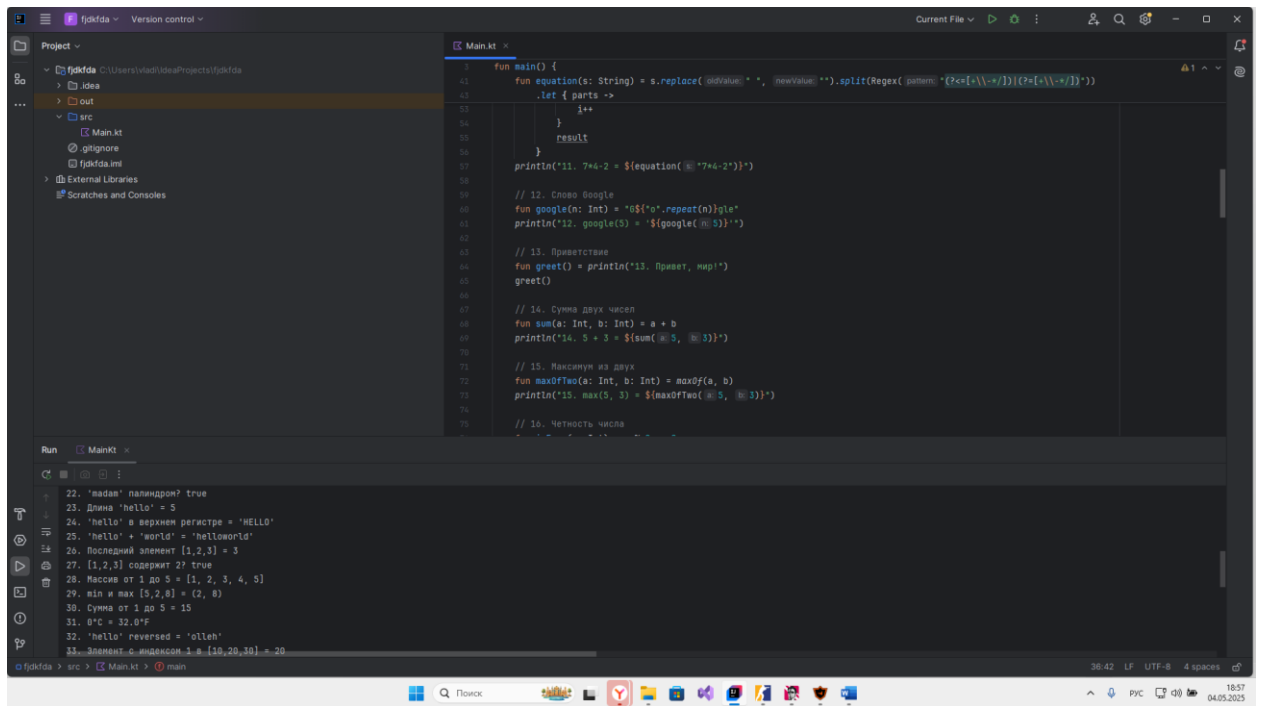
```
1 5 * 8 = 40
2 5 * 9 = 45
3 0 + 10 = 10
4 35. Делится 'test' = 4
5 39. [1,2,3] reversed = [3, 2, 1]
6 40. Копия [1,2,3] + [1, 2, 3]
7 41. Гласных в 'hello' = 2
8 42. Индекс 2 в [1,2,3,2] = 1
9
10 Process finished with exit code 0
```

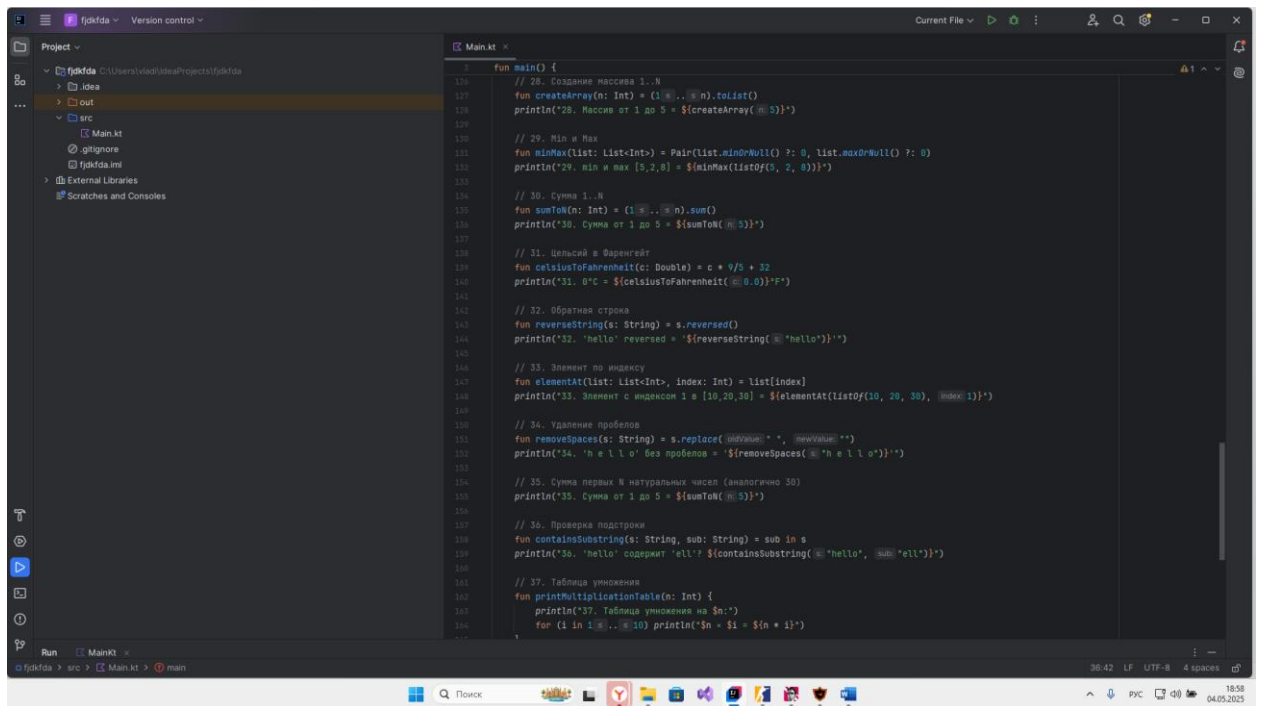
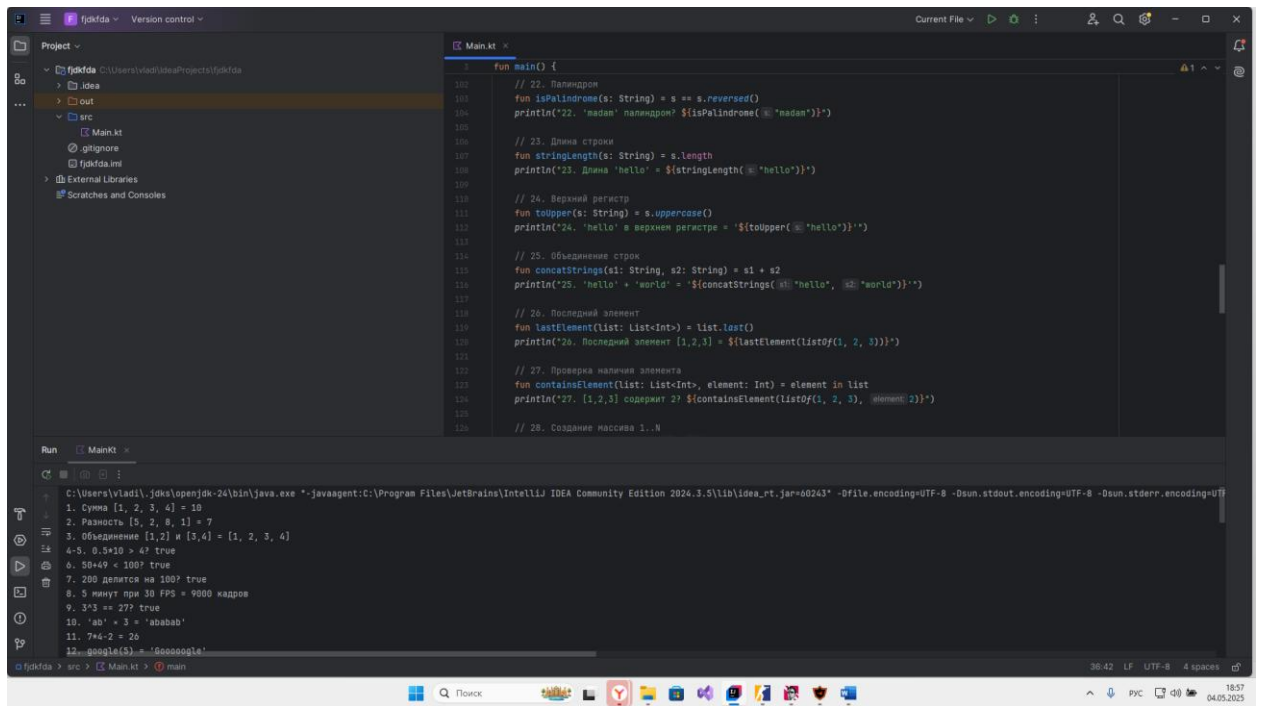


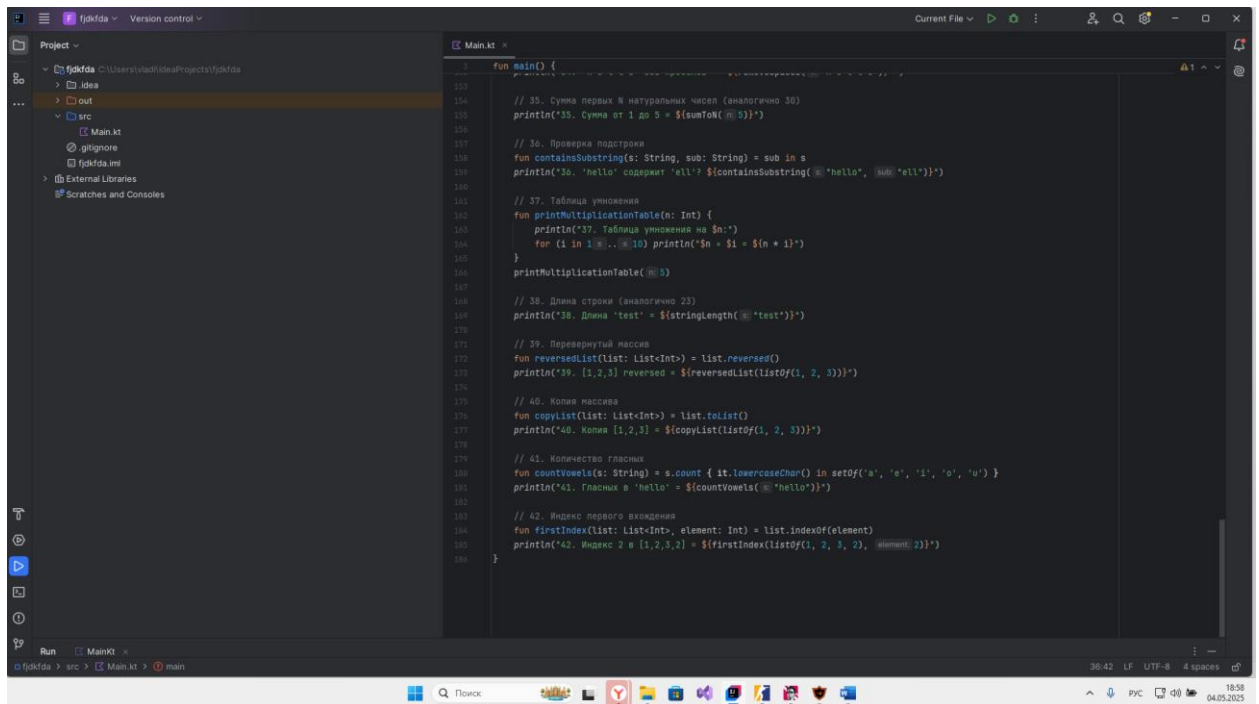
```
1 fun main() {
2
3     // 8. Количество кадров
4     fun frames(minutes: Int, fps: Int) = minutes * 60 * fps
5     println("8. 5 минут при 30 FPS = ${frames(minutes=5, fps=30)} кадров")
6
7     // 9. Проверка k*n == n
8     fun kTok(k: Int, n: Int) = k.toDouble().pow(k) == n.toDouble()
9     println("9. 3^3 == 27? ${kTok(3, 27)}")
10
11    // 10. Повторение строки (рекурсивно)
12    fun repetition(txt: String, n: Int): String = if (n == 0) "" else txt + repetition(txt, n - 1)
13    println("10. 'ab' * 3 = ${repetition("ab", 3)}")
14
15    // 11. Решение уравнения (упрощенная версия)
16    fun equation(s: String) = s.replace(Regex("ab", "newvalue"), "newvalue").split(Regex("(?<=[+\\-*/])(?=[+\\-*/])"))
17    .filter { it.isNotBlank() }
18    .let { parts ->
19        var result = parts[0].toInt()
20        var i = 1
21        while (i < parts.size) {
22            when (parts[i]) {
23                "+" -> result += parts[i+1].toInt()
24                "-" -> result -= parts[i+1].toInt()
25                "*" -> result *= parts[i+1].toInt()
26                "/" -> result /= parts[i+1].toInt()
27            }
28            i++
29        }
30    }
31}
```

Run Main.kt

```
1 34. 'h e l l o' без пробелов = 'hello'
2 35. Сумма от 1 до 5 = 15
3 36. 'hello' содержит 'ell'? true
4 37. Таблица умножения на 5:
5 5 * 1 = 5
6 5 * 2 = 10
7 5 * 3 = 15
8 5 * 4 = 20
9 5 * 5 = 25
10 5 * 6 = 30
11 5 * 7 = 35
12 5 * 8 = 40
```







import kotlin.math.pow

fun main() {

// 1. Сумма элементов списка

fun sumList(list: List<Int>) = list.sum()

println("1. Сумма [1, 2, 3, 4] = \${sumList(listOf(1, 2, 3, 4))}")

// 2. Разность max и min

fun diffMaxMin(list: List<Int>) = if (list.isEmpty()) 0 else list.max() - list.min()

println("2. Разность [5, 2, 8, 1] = \${diffMaxMin(listOf(5, 2, 8, 1))}")

// 3. Объединение списков

fun combineLists(list1: List<Int>, list2: List<Int>) = list1 + list2

println("3. Объединение [1,2] и [3,4] = \${combineLists(listOf(1, 2), listOf(3, 4))}")

// 4-5. Проверка prob*prize > pay

fun isProfitable(prob: Double, prize: Double, pay: Double) = prob * prize > pay

println("4-5. 0.5*10 > 4? \${isProfitable(0.5, 10.0, 4.0)}")

// 6. Сумма < 100

fun sumLessThan100(a: Int, b: Int) = a + b < 100

println("6. 50+49 < 100? \${sumLessThan100(50, 49)}")

// 7. Делится на 100

fun divisibleBy100(n: Int) = n % 100 == 0

println("7. 200 делится на 100? \${divisibleBy100(200)}")

```

// 8. Количество кадров
fun frames(minutes: Int, fps: Int) = minutes * 60 * fps
println("8. 5 минут при 30 FPS = ${frames(5, 30)} кадров")

// 9. Проверка  $k^n == n$ 
fun kToK(k: Int, n: Int) = k.toDouble().pow(k) == n.toDouble()
println("9.  $3^3 == 27$ ? ${kToK(3, 27)}")

// 10. Повторение строки (рекурсивно)
fun repetition(txt: String, n: Int): String = if (n == 0) "" else txt + repetition(txt, n - 1)
println("10. 'ab' × 3 = '${repetition("ab", 3)}'")

// 11. Решение уравнения (упрощенная версия)
fun equation(s: String) = s.replace(" ", "").split(Regex("(?<=[+\\-*/])|(?=[+\\-*/])"))
    .filter { it.isNotBlank() }
    .let { parts ->
        var result = parts[0].toInt()
        var i = 1
        while (i < parts.size) {
            when (parts[i]) {
                "+" -> result += parts[++i].toInt()
                "-" -> result -= parts[++i].toInt()
                "*" -> result *= parts[++i].toInt()
                "/" -> result /= parts[++i].toInt()
            }
            i++
        }
        result
    }
println("11.  $7*4-2 =$  ${equation("7*4-2")}$  $")$ 

// 12. Слово Google
fun google(n: Int) = "G${"o".repeat(n)}gle"
println("12. google(5) = '${google(5)}'")

// 13. Приветствие
fun greet() = println("13. Привет, мир!")
greet()

// 14. Сумма двух чисел
fun sum(a: Int, b: Int) = a + b
println("14. 5 + 3 = ${sum(5, 3)}")

```

```

// 15. Максимум из двух
fun maxOfTwo(a: Int, b: Int) = maxOf(a, b)
println("15. max(5, 3) = ${maxOfTwo(5, 3)}")

// 16. Четность числа
fun isEven(n: Int) = n % 2 == 0
println("16. 4 четное? ${isEven(4)}")

// 17. Факториал
fun factorial(n: Int): Int = if (n <= 1) 1 else n * factorial(n - 1)
println("17. 5! = ${factorial(5)}")

// 18. Проверка на простоту
fun isPrime(n: Int): Boolean {
    if (n <= 1) return false
    for (i in 2 until n) if (n % i == 0) return false
    return true
}
println("18. 7 простое? ${isPrime(7)}")

// 19. Сумма массива (аналогично 1)
println("19. Сумма [1,2,3] = ${sumList(listOf(1, 2, 3))}")

// 20. Максимум в массиве
fun maxInList(list: List<Int>) = list.maxOrNull() ?: 0
println("20. max [1,5,3] = ${maxInList(listOf(1, 5, 3))}")

// 21. Сортировка массива
fun sortList(list: List<Int>) = list.sorted()
println("21. Сортировка [3,1,2] = ${sortList(listOf(3, 1, 2))}")

// 22. Палиндром
fun isPalindrome(s: String) = s == s.reversed()
println("22. 'madam' палиндром? ${isPalindrome("madam")}")

// 23. Длина строки
fun stringLength(s: String) = s.length
println("23. Длина 'hello' = ${stringLength("hello")}")

// 24. Верхний регистр
fun toUpper(s: String) = s.uppercase()
println("24. 'hello' в верхнем регистре = '${toUpper("hello")}'")

```

```

// 25. Объединение строк
fun concatStrings(s1: String, s2: String) = s1 + s2
println("25. 'hello' + 'world' = '${concatStrings("hello", "world")}'")

// 26. Последний элемент
fun lastElement(list: List<Int>) = list.last()
println("26. Последний элемент [1,2,3] = ${lastElement(listOf(1, 2, 3))}")

// 27. Проверка наличия элемента
fun containsElement(list: List<Int>, element: Int) = element in list
println("27. [1,2,3] содержит 2? ${containsElement(listOf(1, 2, 3), 2)}")

// 28. Создание массива 1..N
fun createArray(n: Int) = (1..n).toList()
println("28. Массив от 1 до 5 = ${createArray(5)}")

// 29. Min и Max
fun minMax(list: List<Int>) = Pair(list.minOrNull() ?: 0, list.maxOrNull() ?: 0)
println("29. min и max [5,2,8] = ${minMax(listOf(5, 2, 8))}")

// 30. Сумма 1..N
fun sumToN(n: Int) = (1..n).sum()
println("30. Сумма от 1 до 5 = ${sumToN(5)}")

// 31. Цельсий в Фаренгейт
fun celsiusToFahrenheit(c: Double) = c * 9/5 + 32
println("31. 0°C = ${celsiusToFahrenheit(0.0)}°F")

// 32. Обратная строка
fun reverseString(s: String) = s.reversed()
println("32. 'hello' reversed = '${reverseString("hello")}'")

// 33. Элемент по индексу
fun elementAt(list: List<Int>, index: Int) = list[index]
println("33. Элемент с индексом 1 в [10,20,30] = ${elementAt(listOf(10, 20, 30), 1)}")

// 34. Удаление пробелов
fun removeSpaces(s: String) = s.replace(" ", "")
println("34. 'h e l l o' без пробелов = '${removeSpaces("h e l l o")}'")

// 35. Сумма первых N натуральных чисел (аналогично 30)
println("35. Сумма от 1 до 5 = ${sumToN(5)}")

```

```

// 36. Проверка подстроки
fun containsSubstring(s: String, sub: String) = sub in s
println("36. 'hello' содержит 'ell'? ${containsSubstring("hello", "ell")}")

// 37. Таблица умножения
fun printMultiplicationTable(n: Int) {
    println("37. Таблица умножения на $n:")
    for (i in 1..10) println("$n × $i = ${n * i}")
}
printMultiplicationTable(5)

// 38. Длина строки (аналогично 23)
println("38. Длина 'test' = ${stringLength("test")}")

// 39. Перевернутый массив
fun reversedList(list: List<Int>) = list.reversed()
println("39. [1,2,3] reversed = ${reversedList(listOf(1, 2, 3))}")

// 40. Копия массива
fun copyList(list: List<Int>) = list.toList()
println("40. Копия [1,2,3] = ${copyList(listOf(1, 2, 3))}")

// 41. Количество гласных
fun countVowels(s: String) = s.count { it.lowercaseChar() in setOf('a', 'e', 'i', 'o', 'u') }
println("41. Гласных в 'hello' = ${countVowels("hello")}")

// 42. Индекс первого вхождения
fun firstIndex(list: List<Int>, element: Int) = list.indexOf(element)
println("42. Индекс 2 в [1,2,3,2] = ${firstIndex(listOf(1, 2, 3, 2), 2)}")
}

```