

Первое задание

Операции с числами

A) fun main() {

val number = 42 // Замените это число на любое двузначное число

// Проверка, является ли число двузначным

if (number in 10..99) {

val tens = number / 10 // Число десятков

println("Число десятков: \$tens")

} else {

println(" введите двузначное число.")

}

}

Б) fun main() {

val number = 42 // Замените это число на любое двузначное число

// Проверка, является ли число двузначным

if (number in 10..99) {

val units = number % 10 // Число единиц

println("Число единиц: \$units")

} else {

println("введите двузначное число.")

}

}

В) fun main() {

val number = readLine()?.toIntOrNull() ?: return // Вводим число

if (number !in 10..99) { // Проверяем, является ли оно двузначным

println("Введено неверное значение")

return

}

```

    val sumOfDigits = number / 10 + number % 10 // Находим сумму цифр
    println(sumOfDigits)
}

```

```

Г) fun main() {
    val number = readLine()?.toIntOrNull() ?: return // Вводим число

    if (number !in 10..99) { // Проверяем, является ли оно двузначным
        println("Введено неверное значение")
        return
    }
}

```

```

    val productOfDigits = (number / 10) * (number % 10) // Находим произведение цифр
    println(productOfDigits)
}

```

Второе задание

```

А) fun main() {
    val number = readLine()?.toIntOrNull() ?: return // Вводим число

    if (number !in 100..999) { // Проверяем, является ли оно трехзначным
        println("Введено неверное значение")
        return
    }
}

```

```

    val units = number % 10 // Находим единицы
    println(units)
}

```

```

Б) fun main() {
    val number = readLine()?.toIntOrNull() ?: return // Вводим число
}

```

```
if (number !in 100..999) { // Проверяем, является ли оно трехзначным
    println("Введено неверное значение")
    return
}
```

```
val tens = (number / 10) % 10 // Находим десятки
println(tens)
}
```

В) fun main() {

```
    val number = readLine()?.toIntOrNull() ?: return // Вводим число
```

```
    if (number !in 100..999) { // Проверяем, является ли оно трехзначным
        println("Введено неверное значение")
        return
    }
```

```
    val sumOfDigits = number / 100 + (number / 10) % 10 + number % 10 // Находим сумму цифр
    println(sumOfDigits)
}
```

Г) fun main() {

```
    val number = readLine()?.toIntOrNull() ?: return // Вводим число
```

```
    if (number !in 100..999) { // Проверяем, является ли оно трехзначным
        println("Введено неверное значение")
        return
    }
```

```
    val productOfDigits = (number / 100) * ((number / 10) % 10) * (number % 10) // Находим
    произведение цифр
    println(productOfDigits)
}
```

Третье задание

```
fun main() {  
    print("Введите первое число: ")  
    val num1 = readLine()?.toDoubleOrNull()  
  
    print("Введите второе число: ")  
    val num2 = readLine()?.toDoubleOrNull()  
  
    if (num1 == null || num2 == null) {  
        println("Ошибка: введенные данные должны быть числами.")  
        return  
    }  
  
    if (num2 == 0.0) {  
        println("Ошибка: деление на ноль невозможно.")  
        return  
    }  
  
    val result = num1 / num2  
    println("$num1 поделить на $num2 равно $result")  
}
```

Четвертое задание

```
fun main() {  
    print("Введите основание степени: ")  
    val base = readLine()?.toDoubleOrNull()  
  
    print("Введите показатель степени: ")  
    val exponent = readLine()?.toDoubleOrNull()
```

```
if (base == null || exponent == null) {  
    println("Ошибка: введенные данные должны быть числами.")  
    return  
}  
  
val result = Math.pow(base, exponent)  
println("$base в степени $exponent равно $result")  
}
```

Пятое задание

```
import kotlin.math.sqrt  
  
fun main() {  
    print("Введите число: ")  
    val input = readLine()?.toDoubleOrNull()  
  
    if (input == null) {  
        println("Ошибка: введенное значение должно быть числом.")  
        return  
    }  
  
    if (input < 0) {  
        println("Ошибка: квадратный корень из отрицательного числа не существует в области действительных чисел.")  
        return  
    }  
  
    val root = sqrt(input)  
    println("Квадратный корень из $input равен $root")  
}
```

Вычисление логических выражений

Первое задание

```
A) fun main() {  
    val A = true  
    val B = false  
  
    val result = A || B  
    println(result) // Выведет "true"  
}
```

```
B) fun main() {  
    val A = true  
    val B = false  
  
    val result = A && B  
    println(result) // Выведет "false"  
}
```

```
Г) fun main() {  
    val B = false  
    val C = false  
  
    val result = B || C  
    println(result) // Выведет "false"  
}
```

Второе задание

```
A) fun main() {  
    val X = false  
    val Z = false
```

```
    val result = X || Z
    println(result) // Выведет "false"
}
```

```
B) fun main() {
    val X = false
    val Y = true

    val result = X && Y
    println(result) // Выведет "false"
}
```

```
Г) fun main() {
    val X = false
    val Z = false

    val result = X && Z
    println(result) // Выведет "false"
}
```

Третье задание

```
A) fun main() {
    val A = true
    val B = false

    val result = (!A) && B
    println(result) // Выведет "false"
}
```

```
B) fun main() {
    val A = true
```

```
val B = false
```

```
val result = A || (!B)
```

```
println(result) // Выведет "true"
```

```
}
```

```
Г) fun main() {
```

```
    val A = true
```

```
    val B = false
```

```
    val C = false
```

```
    val result = (A && B) || C
```

```
    println(result) // Выведет "false"
```

```
}
```

Четвертое задание

```
A) fun main() {
```

```
    val X = true
```

```
    val Y = true
```

```
    val result = (!X) && Y
```

```
    println(result) // Выведет "false"
```

```
}
```

```
B) fun main() {
```

```
    val X = true
```

```
    val Y = true
```

```
    val result = X || (!Y)
```

```
    println(result) // Выведет "true"
```

```
}
```



```
Г) fun main() {  
    val X = true  
    val Y = true  
    val Z = false  
  
    val result = X || (Y && Z)  
    println(result) // Выведет "true"  
}
```

Пятое задание

```
А) fun main() {  
    val X = true  
    val Y = true  
  
    val result = (!X) && Y  
    println(result) // Выведет "false"  
}
```

```
В) fun main() {  
    val X = true  
    val Y = true  
  
    val result = X || (!Y)  
    println(result) // Выведет "true"  
}
```

```
Г) fun main() {  
    val X = true  
    val Y = true  
    val Z = false  
  
    val result = X || (Y && Z)
```

```
println(result) // Выведет "true"  
}
```

Шестое задание

A) fun main() {
 val X = false
 val Y = false
 val Z = true

 val result = X || (Y && !Z)
 println(result) // Выведет "false"
}

Г) fun main() {
 val X = false
 val Y = false
 val Z = true

 val result = X && (!Y) || Z
 println(result) // Выведет "true"
}

Б) fun main() {
 val X = false
 val Y = false

 val result = (!X) && (!Y)
 println(result) // Выведет "false"
}

```
Д) fun main() {  
    val X = false  
    val Y = false  
    val Z = true  
  
    val result = X && (!Y || Z)  
    println(result) // Выведет "false"  
}
```

```
В) fun main() {  
    val X = false  
    val Y = false  
    val Z = true  
  
    val result = !(X && Z) || Y  
    println(result) // Выведет "true"  
}
```

```
Е) fun main() {  
    val X = false  
    val Y = false  
    val Z = true  
  
    val result = X || !(Y || Z)  
    println(result) // Выведет "false"  
}
```

Седьмое задание

```
А) fun main() {  
    val A = true  
    val B = false
```

```
val C = false
```

```
val result = A || !(A && B) || C  
println(result) // Выведет "true"  
}
```

```
B) fun main() {
```

```
    val A = true
```

```
    val B = false
```

```
    val C = false
```

```
    val result = !A || A && (B || C)  
    println(result) // Выведет "false"  
}
```

```
Г) fun main() {
```

```
    val A = true
```

```
    val B = false
```

```
    val C = false
```

```
    val result = (A || (B && !C)) && C  
    println(result) // Выведет "false"  
}
```