Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Дисциплина: Программирование мобильных информационных систем

Отчёт

по лабораторной работе №1

на тему

**Основы языка Kotlin**

Выполнил: Проверил:

ст. гр. 214302 Усенко Ф.В.

Кирик Р.С.

Минск 2024

1. Задание: **Определение времени года и дня недели:** Напишите программу, которая принимает дату в формате dd.mm.yyyy и определяет как сезон года, так и день недели. Программа должна учитывать особенности високосных годов и нестандартных календарей.

Листинг кода:

fun main() {

while(true) {

println("Введите дату в формате dd.MM.yyyy:")

val input = readLine()

if (input.isNullOrBlank()) {

println("Некорректный ввод.")

return

}

val parts = input.split(".")

if (parts.size != 3) {

println("Неверный формат даты. Используйте формат dd.MM.yyyy")

return

}

val day = parts[0].toIntOrNull()

val month = parts[1].toIntOrNull()

val year = parts[2].toIntOrNull()

if (day == null || month == null || year == null) {

println("Неверный формат даты. Все части должны быть числами.")

return

}

if (!isValidDate(day, month, year)) {

println("Некорректная дата.")

return

}

val dayOfWeek = getDayOfWeek(day, month, year)

val dayOfWeekName = getDayOfWeekName(dayOfWeek)

val season = getSeason(month)

println("День недели: $dayOfWeekName")

println("Сезон года: $season")

}

}

fun isValidDate(day: Int, month: Int, year: Int): Boolean {

if (month < 1 || month > 12) return false

if (day < 1) return false

val daysInMonth = when (month) {

1, 3, 5, 7, 8, 10, 12 -> 31

4, 6, 9, 11 -> 30

2 -> if (isLeapYear(year)) 29 else 28

else -> return false

}

return day <= daysInMonth

}

fun getDayOfWeek(day: Int, month: Int, year: Int): Int {

var adjustedYear = year

var adjustedMonth = month

if (month < 3) {

adjustedYear--

adjustedMonth += 10

} else {

adjustedMonth -= 2

}

val result = (day + 31 \* adjustedMonth / 12 + adjustedYear + adjustedYear / 4 - adjustedYear / 100 + adjustedYear / 400) % 7

return if (result < 0) result + 7 else result

}

fun isLeapYear(year: Int): Boolean {

return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)

}

fun daysInMonth(month: Int, year: Int): Int {

return when (month) {

1, 3, 5, 7, 8, 10, 12 -> 31

4, 6, 9, 11 -> 30

2 -> if (isLeapYear(year)) 29 else 28

else -> throw java.lang.Exception("Ошибка")

}

}

fun getDayOfWeekName(dayOfWeek: Int): String {

return when (dayOfWeek) {

0 -> "Воскресенье"

1 -> "Понедельник"

2 -> "Вторник"

3 -> "Среда"

4 -> "Четверг"

5 -> "Пятница"

6 -> "Суббота"

else -> "Неопределен"

}

}

fun getSeason(month: Int): String {

return when (month) {

12, 1, 2 -> "Зима"

3, 4, 5 -> "Весна"

6, 7, 8 -> "Лето"

9, 10, 11 -> "Осень"

else -> "Неопределен"

}

}

Контрольные вопросы:

1. **Как организован ввод данных с консоли в *Kotlin*? Чем отличаются функции *print*() и *println*()?**

Для вывода данных на консоль используются функции *print*() и *println*(). Функция *println*() добавляет перевод строки после вывода текста, *print*() – нет.

1. **Какие операторы используются для сравнения чисел в Kotlin?**
   1. ==, != — проверка на равенство и неравенство;
   2. >, < — больше и меньше;
   3. >=, <= — больше или равно, меньше или равно.
2. **Чем отличаются переменные, объявленные с использованием *val* и *var*?**

Переменные, объявленные с помощью *val* являются неизменяемыми переменными. То есть мы можем присвоить значение такой переменной только один раз, но изменить его после первого присвоения мы уже не сможем. В свою очередь переменные, объявленные с помощью *var* имеют возможность многократно менять значения.

**4. Как объявить строковую переменную в Kotlin? Можно ли её изменить после объявления?**

Для объявления строковой переменной в Kotlin используется тип String. Пример:

val immutableString: String = "Hello, Kotlin!" // Нельзя изменить

var mutableString: String = "Hello, Kotlin!" // Можно изменить

mutableString = "New Value" // Это допустимо для var

* Переменную val нельзя изменить после объявления (она неизменяемая).
* Переменную var можно изменить после объявления.

**5. Что такое интерполяция строк, и как она работает в Kotlin?**

Интерполяция строк позволяет вставлять значения переменных или выражения прямо в строку. Для этого используется символ $.

Пример:

val name = "Kotlin"

val age = 7

println("Язык $name существует $age лет.")

println("Через 3 года ему будет ${age + 3} лет.") // Вложенное выражение

**6. Как в Kotlin можно обработать возможность появления null?**

Kotlin поддерживает безопасность от null. Для этого используются:

1 Объявление типов с ? для разрешения значения null.

2 Оператор безопасного вызова ?. для обращения к объекту, который может быть null.

3 Оператор ?: для задания значения по умолчанию.

4 Конструкция !! для явного указания, что значение не равно null (может вызвать исключение).

Пример:

val nullableString: String? = null

println(nullableString?.length ?: "Значение равно null") // Безопасный вызов

**7. Как организован ввод данных с консоли в Kotlin?**

Для ввода данных с консоли используется функция readLine():

println("Введите имя:")

val name = readLine() // Ввод строки

println("Привет, $name!")

**8. Чем оличаются функции print() и println()?**

* print() выводит текст без перехода на новую строку.
* println() выводит текст и переводит курсор на новую строку.

Пример:

print("Привет")

print("Мир") // ПриветМир

println("Привет")

println("Мир") // Привет\nМир

**9. Как работают арифметические операторы в Kotlin? Приведите примеры.**

Арифметические операторы: +, -, \*, /, %.

Пример:

val a = 10

val b = 3

println(a + b) // 13

println(a - b) // 7

println(a \* b) // 30

println(a / b) // 3

println(a % b) // 1

**10. Какие операторы используются для сравнения чисел в Kotlin?**

Операторы сравнения:

== (равно)

!= (не равно)

<, >, <=, >= (меньше, больше, меньше или равно, больше или равно).

Пример:

val x = 5

val y = 10

println(x < y) // true

println(x == y) // false

println(x != y) // true

**11. Как работает конструкция if...else в Kotlin?**

Конструкция if...else используется для выполнения кода на основе условий.

Пример:

val x = 5

if (x > 0) {

println("Положительное число")

} else {

println("Неположительное число")

}

**12. В чем разница между конструкциями if...else и when?**

if...else используется для проверки логических условий.

when используется для проверки множества значений и замены множества if.

Пример when:

val day = 3

val dayName = when (day) {

1 -> "Понедельник"

2 -> "Вторник"

else -> "Другой день"

}

println(dayName)

**13. Как создать функцию в Kotlin, которая возвращает значение? Приведите пример.**

Функция с возвращаемым значением:

fun square(x: Int): Int {

return x \* x

}

println(square(4)) // 16

**14. Что такое параметры функции по умолчанию, и как они используются в Kotlin?**

Параметры функции по умолчанию задаются значением по умолчанию, если оно не передано.

Пример:

fun greet(name: String = "Гость") {

println("Привет, $name!")

}

greet() // Привет, Гость!

greet("Котлин") // Привет, Котлин!

**15. Можно ли передать функцию в качестве аргумента другой функции? Приведите пример.**

Kotlin можно передать функцию как аргумент:

fun operate(x: Int, y: Int, operation: (Int, Int) -> Int): Int {

return operation(x, y)

}

val result = operate(5, 3) { a, b -> a + b }

println(result) // 8

**16. Как Kotlin обрабатывает ошибки ввода данных, например, когда пользователь вводит строку вместо числа?**

Для обработки ошибок используется try...catch:

try {

println("Введите число:")

val number = readLine()?.toInt() // Преобразование строки в число

println("Вы ввели $number")

} catch (e: NumberFormatException) {

println("Ошибка: введите число!")

}

**17. В чем разница между неизменяемыми и изменяемыми переменными?**

* val — неизменяемая переменная, значение задаётся один раз.
* var — изменяемая переменная, значение можно менять.

Пример:

val immutable = 10 // Нельзя изменить

var mutable = 20

mutable = 30 // Допустимо

**18. Как организовать цикл для вывода всех чисел от 1 до 10? Приведите пример.**

Для этого используется цикл for:

for (i in 1..10) {

println(i)

}

**19. Какую конструкцию следует использовать для выбора действия на основе множества значений переменной?**

Для этого используется конструкция when:

val color = "red"

when (color) {

"red" -> println("Красный")

"green" -> println("Зеленый")

else -> println("Другой цвет")

}

**20. Как организовать проверку, является ли введенное число четным или нечетным? Приведите пример кода.**

Пример:

println("Введите число:")

val number = readLine()?.toIntOrNull()

if (number != null) {

if (number % 2 == 0) {

println("Число четное")

} else {

println("Число нечетное")

}

} else {

println("Ошибка: введите число!")

}