

Сборник заданий для семинарских занятий  
по курсу  
«Разработка мобильных приложений»

# Содержание

<b>1</b>	<b>Общие сведения</b>	<b>3</b>
<b>2</b>	<b>Язык Kotlin</b>	<b>3</b>
2.1	Семинар «Разработка первой программы на Kotlin» (2 часа) . . . . .	4
2.2	Практическая работа «Функции Kotlin» (2 часа) . . . . .	6
2.3	Семинар «Оператор when» (2 часа) . . . . .	9
2.4	Семинары «Функции высшего порядка в Kotlin/Функциональный подход в Kotlin» (4 часа) . . . . .	11
2.5	Семинар «Поддержка ООП в Kotlin» (4 часа) . . . . .	19
<b>3</b>	<b>Семинар для рубежного контроля</b>	<b>22</b>
<b>4</b>	<b>Разработка на платформе Android</b>	<b>22</b>
4.1	Семинар «Простейшие программы для Android» (2 часа) . . . . .	23
4.2	Семинар «MVVM» (2 часа) . . . . .	26
4.3	Семинар «Работа со списками» (4 часа) . . . . .	27
4.4	Семинары «Использование библиотек для поддержки Dependency Injection и Clean Architecture» (2 часа) . . . . .	28
4.5	Семинар «Работа с внешними сервисами RestAPI» (2 часа) . . . . .	29
<b>5</b>	<b>Семинар для подготовки к зачёту</b>	<b>30</b>
<b>6</b>	<b>Проекты как замена заданиям семинаров</b>	<b>30</b>
<b>7</b>	<b>Зачёт</b>	<b>30</b>
<b>8</b>	<b>Курсовая работа</b>	<b>30</b>
<b>9</b>	<b>Список литературы</b>	<b>31</b>

## 1 Общие сведения

Сборник содержит задания для семинарских занятий по курсу «Разработка мобильных приложений». Задачник рассчитан на 32 часа семинарских занятий.

Структурно курс делится на две части:

1. изучение языка Kotlin;
2. изучение разработки на платформе Android.

В практических и лабораторных работах предполагается использование языка Kotlin в средах IntelliJ IDEA и Android Studio. При этом необходимо соблюдать [Coding Conventions](#) (требования к стилю кода).

Перед сдачей работы добейтесь, чтобы среда не выдавала предупреждений при запуске подпункта **Inspect Code** пункта меню **Analyze**.

Задачи в большей степени рассчитаны на освоение возможностей языка, а не на алгоритмические сложности. Поэтому осуществляйте написание кода в соответствии с заданием, а не с целью, чтобы он просто работал.

Обратите внимание на то, что во всех заданиях необходимо проверять корректность входных данных: программа не должна *падать* ни в каких ситуациях.

## 2 Язык Kotlin

## 2.1 Семинар «Разработка первой программы на Kotlin» (2 часа)

Цель первых четырёх семинаров — обеспечить подготовку к выполнению заданий, нацеленных на освоение особенностей языка Kotlin. В первой работе необходимо написать небольшую программу, чтобы убедиться в понимании базовых конструкций языка (функция main, ветвления, циклы) и целого типа, а также обеспечить успешную настройку среды разработки.

При выполнении заданий обращайтесь внимание на использование специфических особенностей языка везде, где это возможно: if и when могут быть как операторами, так и частью выражений; фигурные скобки во многих случаях можно опускать; точки с запятой почти никогда не используются, корректно выбирайте, как помечать переменные: ключевым словом var или val.

При выполнении работ обеспечивайте **оптимальность** предлагаемой программы как по скорости, так и по памяти. В случае противоречия между двумя критериями, выбирайте алгоритм, который обеспечивает лучшее быстроедействие.

В частности, это обозначает, что:

1. Ввод осуществляется как строка, которую необходимо преобразовать в целое число.
2. Далее вся работа должна вестись **исключительно с числом**, используя только арифметические операции (остаток от деления, целочисленное деление, циклы, условия).
3. Использование строковых методов (включая преобразование числа обратно в строку) **запрещено**.
4. Следует избегать сложных структур данных (списков, множеств, массивов).

### Задание

Для данного неотрицательного целого числа (в пределах Int) найдите указанный результат. Осуществите проверку корректности ввода. Оформите программу и устраните все warning.

1. Сумму чётных цифр
2. Сумму нечётных цифр
3. Произведение чётных цифр
4. Произведение нечётных цифр
5. Максимальную чётную цифру
6. Минимальную чётную цифру
7. Максимальную нечётную цифру
8. Минимальную нечётную цифру
9. Сумму цифр, кратных трём
10. Сумму цифр, не кратных трём
11. Произведение цифр, кратных трём
12. Произведение цифр, не кратных трём
13. Максимальную цифру, кратную трём
14. Минимальную цифру, кратную трём
15. Максимальную цифру, не кратную трём
16. Минимальную цифру, не кратную трём
17. Сумму цифр, стоящих на чётных позициях в числе (если нумеровать цифры с конца): для числа 1234 — ответ 4
18. Произведение цифр, стоящих на чётных позициях в числе (если нумеровать цифры с конца): для числа 1234 — ответ 3

19. Максимальную цифру среди стоящих на чётных позициях в числе (если нумеровать цифры с конца): для числа 1234 — ответ 3
20. Минимальную цифру среди стоящих на чётных позициях в числе (если нумеровать цифры с конца): для числа 1234 — ответ 1
21. Сумму цифр, стоящих на позициях в числе, номера которых кратны трём (если нумеровать цифры с конца): для числа 1234 — ответ 2
22. Произведение цифр, стоящих на позициях в числе, номера которых кратны трём (если нумеровать цифры с конца): для числа 1234 — ответ 2
23. Максимальную цифру среди стоящих на позициях в числе, номера которых кратны трём (если нумеровать цифры с конца): для числа 1234 — ответ 2
24. Минимальную цифру среди стоящих на позициях в числе, номера которых кратны трём (если нумеровать цифры с конца): для числа 1234 — ответ 2
25. Максимальную цифру среди стоящих на позициях в числе, номера которых кратны четырём (если нумеровать цифры с конца): для числа 1234 — ответ 1

## 2.2 Практическая работа «Функции Kotlin» (2 часа)

### Задание №1 (функции и лямбда-функции)

Разработайте программу в соответствии с вашим вариантом. При этом:

- Основной алгоритм (без ввода-вывода) оформите в виде функции.
  - В функцию передавайте исходное число и условие отбора цифр в виде лямбда-функции (например, проверка на чётность, нечётность, кратность трём, не кратность трём — в зависимости от варианта).
  - Функция должна корректно работать при замене условия отбора — то есть быть универсальной.
1. Сумму чётных цифр
  2. Сумму нечётных цифр
  3. Произведение чётных цифр
  4. Произведение нечётных цифр
  5. Максимальную чётную цифру
  6. Минимальную чётную цифру
  7. Максимальную нечётную цифру
  8. Минимальную нечётную цифру
  9. Сумму цифр, кратных трём
  10. Сумму цифр, не кратных трём
  11. Произведение цифр, кратных трём
  12. Произведение цифр, не кратных трём
  13. Максимальную цифру, кратную трём
  14. Минимальную цифру, кратную трём
  15. Максимальную цифру, не кратную трём
  16. Минимальную цифру, не кратную трём
  17. Сумму цифр, стоящих на чётных позициях в числе (если нумеровать цифры с конца): для числа 1234 — ответ 4
  18. Произведение цифр, стоящих на чётных позициях в числе (если нумеровать цифры с конца): для числа 1234 — ответ 3
  19. Максимальную цифру среди стоящих на чётных позициях в числе (если нумеровать цифры с конца): для числа 1234 — ответ 3
  20. Минимальную цифру среди стоящих на чётных позициях в числе (если нумеровать цифры с конца): для числа 1234 — ответ 1
  21. Сумму цифр, стоящих на позициях в числе, номера которых кратны трём (если нумеровать цифры с конца): для числа 1234 — ответ 2
  22. Произведение цифр, стоящих на позициях в числе, номера которых кратны трём (если нумеровать цифры с конца): для числа 1234 — ответ 2
  23. Максимальную цифру среди стоящих на позициях в числе, номера которых кратны трём (если нумеровать цифры с конца): для числа 1234 — ответ 2
  24. Минимальную цифру среди стоящих на позициях в числе, номера которых кратны трём (если нумеровать цифры с конца): для числа 1234 — ответ 2

25. Максимальную цифру среди стоящих на позициях в числе, номера которых кратны четырём (если нумеровать цифры с конца): для числа 1234 — ответ 1

### Задание №2 (функции высшего порядка)

1. Создайте функцию, которая принимает произвольное количество функций с параметром типа `Int` и возвращаемым значением типа `Int`, и возвращает новую функцию — сумму исходных. Для аргумента  $x$  результирующая функция возвращает сумму  $f_1(x) + f_2(x) + \dots + f_n(x)$ .
2. Создайте функцию, которая принимает произвольное количество функций с параметром типа `Int` и возвращаемым значением типа `Int`, и возвращает новую функцию — произведение исходных. Для аргумента  $x$  результирующая функция возвращает произведение  $f_1(x) \cdot f_2(x) \cdot \dots \cdot f_n(x)$ .
3. Создайте функцию, которая принимает произвольное количество функций с параметром типа `Int` и возвращаемым значением типа `Int`, и возвращает новую функцию — максимум исходных. Для аргумента  $x$  результирующая функция возвращает  $\max(f_1(x), f_2(x), \dots, f_n(x))$ .
4. Создайте функцию, которая принимает произвольное количество функций с параметром типа `Int` и возвращаемым значением типа `Int`, и возвращает новую функцию — минимум исходных. Для аргумента  $x$  результирующая функция возвращает  $\min(f_1(x), f_2(x), \dots, f_n(x))$ .
5. Создайте функцию, которая по данной функции  $f : \text{Int} \rightarrow \text{Int}$  и целому числу  $n$  возвращает функцию, вычисляющую  $f$ , применённую  $n$  раз:  $f(f(\dots f(x) \dots))$ .
6. Создайте функцию, которая принимает произвольное количество функций без параметров, возвращающих `String`, и возвращает новую функцию без параметров, возвращающую конкатенацию результатов исходных функций.
7. Создайте функцию, которая принимает произвольное количество функций с параметром типа `Int` и возвращаемым значением типа `Int`, и возвращает новую функцию с аргументом  $x$  типа `Int`, возвращающую **индекс первой** функции (начиная с 0), имеющей **максимальное** значение при подстановке  $x$ .
8. Создайте функцию, которая принимает произвольное количество функций с параметром типа `Int` и возвращаемым значением типа `Int`, и возвращает новую функцию с аргументом  $x$  типа `Int`, возвращающую **индекс первой** функции, имеющей **минимальное** значение при подстановке  $x$ .
9. Создайте функцию, которая принимает произвольное количество функций с параметром типа `Int` и возвращаемым значением типа `Int`, и возвращает новую функцию с аргументом  $x$  типа `Int`, возвращающую **индекс последней** функции, имеющей **максимальное** значение при подстановке  $x$ .
10. Создайте функцию, которая принимает произвольное количество функций с параметром типа `Int` и возвращаемым значением типа `Int`, и возвращает новую функцию с аргументом  $x$  типа `Int`, возвращающую **индекс последней** функции, имеющей **минимальное** значение при подстановке  $x$ .
11. Создайте функцию, которая по двум функциям с параметром типа `Int` и возвращаемым значением типа `Int`? возвращает новую функцию — сумму исходных. Если хотя бы одна из функций возвращает `null`, результат также должен быть `null`.
12. Создайте функцию, которая по двум функциям с параметром типа `Int` и возвращаемым значением типа `Int`? возвращает новую функцию — произведение исходных. Если хотя бы одна из функций возвращает `null`, результат также должен быть `null`.
13. Создайте функцию, которая по двум функциям с параметром типа `Int` и возвращаемым значением типа `Int`? возвращает новую функцию — максимум исходных. Если хотя бы одна из функций возвращает `null`, результат также должен быть `null`.
14. Создайте функцию, которая по двум функциям с параметром типа `Int` и возвращаемым значением типа `Int`? возвращает новую функцию — минимум исходных. Если хотя бы одна из функций возвращает `null`, результат также должен быть `null`.

15. Создайте функцию, которая по двум функциям  $f$  и  $g$  (обе:  $\text{Int} \rightarrow \text{Int}?$ ) возвращает функцию, вычисляющую композицию  $f(g(x))$ . Если  $g(x) = \text{null}$ , то результат —  $\text{null}$ .
16. Создайте функцию, которая по функции  $f : \text{Int} \rightarrow \text{Int}$  и целому числу  $n$  возвращает новую функцию, принимающую массив из  $n$  элементов типа  $\text{Int}$  и возвращающую массив результатов применения  $f$  к каждому элементу.
17. Создайте функцию, которая по данному массиву целых чисел возвращает функцию без параметров, которая при каждом вызове последовательно возвращает элементы массива. После исчерпания элементов возвращает  $\text{null}$ .
18. Создайте функцию, которая по функции  $f : \text{Int} \rightarrow \text{Int}$  возвращает функцию без параметров, которая при каждом вызове последовательно возвращает  $f(1)$ ,  $f(2)$ ,  $f(3)$ , и так далее.
19. Создайте функцию, которая по данному массиву целых чисел возвращает функцию без параметров, которая при каждом вызове последовательно возвращает элементы массива в обратном порядке. После исчерпания элементов возвращает  $\text{null}$ .
20. Создайте функцию, которая по данной строке возвращает функцию без параметров, которая при каждом вызове последовательно возвращает символы строки. После исчерпания символов возвращает  $\text{null}$ .
21. Создайте функцию, которая по данной строке возвращает функцию без параметров, которая при каждом вызове последовательно возвращает символы строки в обратном порядке. После исчерпания символов возвращает  $\text{null}$ .
22. Создайте функцию, которая принимает произвольное количество функций с параметром типа  $\text{Float}$  и возвращаемым значением типа  $\text{Float}$ , и возвращает новую функцию — среднее арифметическое исходных:  $\frac{f_1(x)+f_2(x)+\dots+f_n(x)}{n}$ .
23. Создайте функцию, которая принимает произвольное количество функций с параметром типа  $\text{Float}$  и возвращаемым значением типа  $\text{Float}$ , и возвращает новую функцию — среднее квадратическое исходных:  $\sqrt{\frac{f_1(x)^2+f_2(x)^2+\dots+f_n(x)^2}{n}}$ .
24. Создайте функцию, которая принимает произвольное количество функций с параметром типа  $\text{Float}$  и возвращаемым значением типа  $\text{Float}$ , и возвращает новую функцию — среднее геометрическое исходных:  $\sqrt[n]{f_1(x) \cdot f_2(x) \cdot \dots \cdot f_n(x)}$ . Предполагается, что все значения положительны.
25. Создайте функцию, которая по функции  $f : \text{Float} \rightarrow \text{Float}$  и числу  $x$  возвращает функцию без параметров, которая при каждом вызове последовательно возвращает  $f(x)$ ,  $f(f(x))$ ,  $f(f(f(x)))$ , и так далее.

### Задание №3 (хвостовая рекурсия)

Создайте функцию из задания №1 с использованием хвостовой рекурсии

## 2.3 Семинар «Оператор when» (2 часа)

В работе предполагается освоение оператора when.

При выполнении заданий обращайтесь внимание на использование специфических особенностей языка везде, где это возможно: if и when могут быть как операторами, так и частью выражений; фигурные скобки во многих случаях можно опускать; точки с запятой почти никогда не используются, корректно выбирайте, как помечать переменные: ключевым словом var или val.

При выполнении работ обеспечивайте **оптимальность** предлагаемой программы как по скорости, так и по памяти. В случае противоречия между двумя критериями, выбирайте алгоритм, который обеспечивает лучшее быстроедействие.

В частности, это обозначает, что нельзя использовать дополнительные строки (в заданиях, кроме как для ввода, строки не нужны), следует избегать сложных структур (списков, множеств).

*В задачах предполагается, что не используются специальные функции, предназначенные для работы с датами.*

1. По введённой дате определите дату следующего дня. Выведите её и проверьте, совпадает ли количество дней в месяце исходной даты с количеством дней в месяце полученной даты.
2. По введённой дате определите дату предыдущего дня. Выведите её и проверьте, совпадает ли количество дней в месяце исходной даты с количеством дней в месяце полученной даты.
3. По введённой дате определите дату, которая наступит ровно через месяц (прибавить 1 к месяцу, при необходимости корректируя год). Если в следующем месяце нет дня с таким же числом (например, 31 апреля), то возьмите последний день следующего месяца. Выведите полученную дату и проверьте, является ли она последним днём месяца.
4. По введённой дате определите дату, которая была ровно месяц назад (вычесть 1 из месяца, при необходимости корректируя год). Если в предыдущем месяце нет дня с таким же числом, возьмите последний день предыдущего месяца. Выведите полученную дату и проверьте, является ли она первым днём месяца.
5. По введённой дате определите дату, которая наступит через 2 месяца (прибавить 2 к месяцу, корректируя год). Корректировка дня, как в предыдущих задачах. Выведите полученную дату и проверьте, находится ли она в том же квартале года, что и исходная дата. (Кварталы: 1-3, 4-6, 7-9, 10-12)
6. По введённой дате определите дату, которая была 3 месяца назад. Выведите полученную дату и проверьте, находится ли она в том же году, что и исходная дата.
7. По введённой дате определите дату, которая наступит через 1 год (прибавить 1 к году). Учтите високосность года для февраля. Если исходная дата - 29 февраля, то в следующем невисокосном году возьмите 28 февраля. Выведите полученную дату и проверьте, является ли она високосным днём (29 февраля).
8. По введённой дате определите дату, которая была 1 год назад. Выведите полученную дату и проверьте, была ли исходная дата високосным днём (29 февраля), а полученная - нет.
9. По введённой дате определите дату, которая наступит через 100 дней. Выведите её и проверьте, является ли полученная дата последним днём месяца.
10. По введённой дате определите дату, которая была 100 дней назад. Выведите её и проверьте, является ли полученная дата первым днём месяца.
11. По введённой дате определите дату, которая наступит через 1 неделю (7 дней). Выведите её и проверьте, находится ли полученная дата в том же месяце, что и исходная.
12. По введённой дате определите дату, которая была 1 неделю назад. Выведите её и проверьте, находится ли полученная дата в том же году, что и исходная.
13. По введённой дате определите дату, которая наступит через 2 месяца. Выведите её и проверьте, является ли день полученной даты последним днём месяца.
14. По введённой дате определите дату, которая была 2 месяца назад. Выведите её и проверьте, является ли день полученной даты первым днём месяца.

15. По введённой дате определите дату, которая наступит через 6 месяцев. Выведите её и проверьте, находится ли полученная дата во второй половине года (месяц с июля по декабрь).
16. По введённой дате определите дату, которая была 6 месяцев назад. Выведите её и проверьте, находится ли полученная дата в первом полугодии (месяц с января по июнь).
17. По введённой дате определите дату, которая наступит через 1 месяц и 1 день (сначала прибавить месяц, затем день). Корректировка дня, как в задаче 3. Выведите полученную дату и проверьте, является ли она первым днём месяца.
18. По введённой дате определите дату, которая была 1 месяц и 1 день назад (сначала вычтеть месяц, затем день). Выведите полученную дату и проверьте, является ли она последним днём месяца.
19. По введённой дате определите дату, которая наступит через 2 года. Выведите её и проверьте, является ли год полученной даты високосным.
20. По введённой дате определите дату, которая была 2 года назад. Выведите её и проверьте, был ли год полученной даты високосным.
21. По введённой дате определите дату, которая наступит через 1 квартал (3 месяца). Выведите её и проверьте, является ли полученная дата последним днём квартала (31 марта, 30 июня, 30 сентября, 31 декабря).
22. По введённой дате определите дату, которая была 1 квартал назад. Выведите её и проверьте, является ли полученная дата первым днём квартала (1 января, 1 апреля, 1 июля, 1 октября).
23. По введённой дате определите дату, которая наступит через 1 год и 1 месяц. Выведите её и проверьте, является ли день полученной даты первым числом месяца.
24. По введённой дате определите дату, которая была 1 год и 1 месяц назад. Выведите её и проверьте, является ли день полученной даты последним числом месяца.
25. По введённой дате определите дату, которая наступит через 366 дней (чтобы перепрыгнуть через год). Выведите её и проверьте, является ли полученная дата високосным днём (29 февраля).

## 2.4 Семинары «Функции высшего порядка в Kotlin/Функциональный подход в Kotlin» (4 часа)

В данной работе требуется написать не самую оптимальную реализацию, а реализацию, которая наиболее полноценно использует функции над коллекциями, использующие функциональный подход, и строковые функции.

В работе запрещено использовать mutable коллекции и var переменные.

Реализация должна состоять из одной строки с точечными вызовами, включая ввод и вывод, использовать рекурсию запрещено.

Примечание: данный способ реализации программы нужен исключительно в учебных целях, в дальнейшем разбивайте подобные решения на небольшие функции, которые удобно повторно использовать.

Не забывайте проверять все входные данные от пользователя полноценно и всесторонне.

### Задание №1

Для данного неотрицательного целого числа (в пределах Int) найдите указанный результат. Осуществите проверку корректности ввода. Оформите программу и уберите все warning.

1. Сумма чётных цифр
2. Сумма нечётных цифр
3. Произведение чётных цифр
4. Произведение нечётных цифр
5. Максимальная чётная цифра
6. Минимальная чётная цифра
7. Максимальная нечётная цифра
8. Минимальная нечётная цифра
9. Сумма цифр, кратных трём
10. Сумма цифр, не кратных трём
11. Произведение цифр, кратных трём
12. Произведение цифр, не кратных трём
13. Максимальная цифра, кратная трём
14. Минимальная цифра, кратная трём
15. Максимальная цифра, не кратная трём
16. Минимальная цифра, не кратная трём
17. Сумма цифр, стоящих на чётных позициях в числе (если пронумеровать цифры с конца): для числа 1234 ответ 4
18. Произведение цифр, стоящих на чётных позициях в числе (если пронумеровать цифры с конца): для числа 1234 ответ 3
19. Максимальная цифра среди стоящих на чётных позициях в числе (если пронумеровать цифры с конца): для числа 1234 ответ 3
20. Минимальная цифра среди стоящих на чётных позициях в числе (если пронумеровать цифры с конца): для числа 1234 ответ 1
21. Сумма цифр, стоящих на позициях в числе, номера которых кратны трём (если пронумеровать цифры с конца): для числа 1234 ответ 2
22. Произведение цифр, стоящих на позициях в числе, номера которых кратны трём (если пронумеровать цифры с конца): для числа 1234 ответ 2

23. Максимальная цифра среди стоящих на позициях в числе, номера которых кратны трём (если нумеровать цифры с конца): для числа 1234 ответ 2
24. Минимальная цифра среди стоящих на позициях в числе, номера которых кратны трём (если нумеровать цифры с конца): для числа 1234 ответ 2
25. Максимальная цифра среди стоящих на позициях в числе, номера которых кратны четырём (если нумеровать цифры с конца): для числа 1234 ответ 1

### **Задание №2**

Можно предполагать, что все символам соответствует одно значение типа char (однако, если вы сделаете корректное решение (разумеется, вне пары), то это будет плюсом).

1. Найдите первый символ в первом максимально длинном слове с нечётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
2. Найдите последний символ в первом максимально длинном слове с нечётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
3. Найдите первый символ в последнем максимально длинном слове с нечётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
4. Найдите последний символ в последнем максимально длинном слове с нечётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
5. Найдите первый символ в первом самом коротком слове в строке с нечётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
6. Найдите последний символ в первом самом коротком слове в строке с нечётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
7. Найдите первый символ в последнем самом коротком слове в строке с нечётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
8. Найдите последний символ в последнем самом коротком слове в строке с нечётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
9. Найдите первый символ в первом максимально длинном слове с чётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
10. Найдите последний символ в первом максимально длинном слове с чётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
11. Найдите первый символ в последнем максимально длинном слове с чётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
12. Найдите последний символ в последнем максимально длинном слове с чётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
13. Найдите первый символ в первом самом коротком слове в строке с чётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
14. Найдите последний символ в первом самом коротком слове в строке с чётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
15. Найдите первый символ в последнем самом коротком слове в строке с чётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
16. Найдите последний символ в последнем самом коротком слове в строке с чётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).

17. Найдите второй символ в первом максимально длинном слове с чётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
18. Найдите предпоследний символ в первом максимально длинном слове с чётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
19. Найдите второй символ в последнем максимально длинном слове с чётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
20. Найдите предпоследний символ в последнем максимально длинном слове с чётным числом символов в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).
21. Найдите второй символ в первом самом коротком слове в строке с чётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
22. Найдите предпоследний символ в первом самом коротком слове в строке с чётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
23. Найдите второй символ в последнем самом коротком слове в строке с чётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
24. Найдите предпоследний символ в последнем самом коротком слове в строке с чётным числом символов (в строке указываются только слова, разделённые одним или несколькими пробелами).
25. Найдите первый символ в первом максимально длинном слове с числом символов, кратным трём, в строке (в строке указываются только слова, разделённые одним или несколькими пробелами).

### Задание №3

1. С клавиатуры вводится описание массива из 10 элементов в виде:  
*номер:значение*  
однако, порядок указания элементов может быть любой. Выведите все элементы массива в порядке возрастания номеров.
2. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какие цифры присутствуют в каждом числе?
3. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какие цифры присутствуют хотя бы в двух числах?
4. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какие цифры присутствуют ровно в одном числе?
5. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какие цифры присутствуют ровно в двух числах?
6. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какие цифры отсутствуют ровно в двух числах?
7. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какие цифры отсутствуют ровно в одном числе?
8. В строке указано несколько слов, разделённых пробелами (по одному пробелу между словами). Какие символы присутствуют в каждом слове?
9. В строке указано несколько слов, разделённых пробелами (по одному пробелу между словами). Какие символы присутствуют хотя бы в двух словах?
10. В строке указано несколько слов, разделённых пробелами (по одному пробелу между словами). Какие символы присутствуют ровно в одном слове?

11. В строке указано несколько слов, разделённых пробелами (по одному пробелу между словами). Какие символы отсутствуют ровно в одном слове?
12. В строке указано несколько слов, разделённых пробелами (по одному пробелу между словами). Какие символы отсутствуют ровно в двух словах?
13. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какое количество чисел удовлетворяет условию отсутствия повторяющихся цифр?
14. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какое количество чисел удовлетворяет условию наличия повторяющихся цифр?
15. В строке указано несколько слов, разделённых пробелами (по одному пробелу между словами). Какое количество слов удовлетворяет условию отсутствия повторяющихся символов?
16. В строке указано несколько слов, разделённых пробелами (по одному пробелу между словами). Какое количество слов удовлетворяет условию наличия повторяющихся символов?
17. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). В каком количестве чисел присутствуют все цифры от 0 до 9?
18. Имеется некоторая последовательность цифр от 0 до 9. С клавиатуры вводится 9 строк следующего вида:  
*цифра->цифра*  
 Каждая строка обозначает, что после цифры, стоящей до стрелки, в последовательности стоит цифра, стоящая после стрелки.  
 Выведите исходную последовательность.
19. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какие цифры присутствуют в каждом числе дважды?
20. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какие цифры присутствуют хотя бы в одном числе дважды?
21. В строке указано несколько неотрицательных целых чисел, разделённых пробелами (по одному пробелу между числами). Какие цифры присутствуют ровно в одном числе дважды?
22. С клавиатуры вводится несколько строк, последняя строка — пустая (пустая строка — признак окончания ввода и дальше игнорируется). Выведите символы, что присутствуют в каждой строке.
23. С клавиатуры вводится несколько строк, последняя строка — пустая (пустая строка — признак окончания ввода и дальше игнорируется). Выведите символы, что присутствуют ровно в одной строке.
24. С клавиатуры вводится несколько строк, последняя строка — пустая (пустая строка — признак окончания ввода и дальше игнорируется). Выведите символы, что присутствуют ровно в двух строках.
25. С клавиатуры вводится несколько строк, последняя строка — пустая (пустая строка — признак окончания ввода и дальше игнорируется). Выведите символы, что отсутствуют ровно в двух строках.

**Задание №4** С клавиатуры вводится несколько целых значений через пробел. Найдите (без учета тех чисел, где соответствующей цифры нет):

1. Побитовое И предпоследней цифры всех чисел
2. Побитовое ИЛИ предпоследней цифры всех чисел
3. Побитовое исключающее ИЛИ предпоследней цифры всех чисел

4. Побитовый штрих Шеффера последней цифры всех чисел (операции выполняются слева направо)
5. Побитовый штрих Шеффера предпоследней цифры всех чисел (операции выполняются слева направо)
6. Побитовая стрелка Пирса последней цифры всех чисел (операции выполняются слева направо)
7. Побитовая стрелка Пирса предпоследней цифры всех чисел (операции выполняются слева направо)
8. Побитовый штрих Шеффера последней цифры всех чисел (операции выполняются справа налево)
9. Побитовый штрих Шеффера предпоследней цифры всех чисел (операции выполняются справа налево)
10. Побитовая стрелка Пирса последней цифры всех чисел (операции выполняются справа налево)
11. Побитовая стрелка Пирса предпоследней цифры всех чисел (операции выполняются справа налево)
12. Побитовое И первой цифры всех чисел
13. Побитовое ИЛИ первой цифры всех чисел
14. Побитовое исключаящее ИЛИ первой цифры всех чисел
15. Побитовое И второй цифры всех чисел
16. Побитовое ИЛИ второй цифры всех чисел
17. Побитовое исключаящее ИЛИ второй цифры всех чисел
18. Побитовый штрих Шеффера первой цифры всех чисел (операции выполняются слева направо)
19. Побитовый штрих Шеффера второй цифры всех чисел (операции выполняются слева направо)
20. Побитовая стрелка Пирса первой цифры всех чисел (операции выполняются слева направо)
21. Побитовая стрелка Пирса второй цифры всех чисел (операции выполняются слева направо)
22. Побитовый штрих Шеффера первой цифры всех чисел (операции выполняются справа налево)
23. Побитовый штрих Шеффера второй цифры всех чисел (операции выполняются справа налево)
24. Побитовая стрелка Пирса первой цифры всех чисел (операции выполняются справа налево)
25. Побитовая стрелка Пирса второй цифры всех чисел (операции выполняются справа налево)

#### **Задание №5**

1. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх лучших студентах по среднему баллу. В случае, если у нескольких студентов средний балл совпадает, то выведите большее число студентов (пока не будут выведены все студенты или не будут полностью исчерпаны студенты с тремя лучшими баллами). Вывод надо осуществлять в порядке убывания среднего балла, а для одинаковых средних баллов — в алфавитном порядке по фамилии и имени.
2. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх лучших студентах по максимальному баллу. В случае, если у нескольких студентов максимальный балл совпадает, то выведите большее число студентов (пока не будут выведены все студенты или не будут полностью исчерпаны студенты с тремя лучшими баллами). Вывод надо осуществлять в порядке убывания максимального балла, а для одинаковых максимальных баллов — в алфавитном порядке по фамилии и имени.

3. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх лучших студентах по минимальному баллу. В случае, если у нескольких студентов минимальный балл совпадает, то выведите большее число студентов (пока не будут выведены все студенты или не будут полностью исчерпаны студенты с тремя лучшими баллами). Вывод надо осуществлять в порядке убывания минимального балла, а для одинаковых минимальных баллов — в алфавитном порядке по фамилии и имени.
4. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх худших студентах по среднему баллу. В случае, если у нескольких студентов средний балл совпадает, то выведите большее число студентов (пока не будут выведены все студенты или не будут полностью исчерпаны студенты с тремя худшими баллами). Вывод надо осуществлять в порядке возрастания среднего балла, а для одинаковых средних баллов — в алфавитном порядке по фамилии и имени.
5. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх худших студентах по максимальному баллу. В случае, если у нескольких студентов максимальный балл совпадает, то выведите большее число студентов (пока не будут выведены все студенты или не будут полностью исчерпаны студенты с тремя худшими баллами). Вывод надо осуществлять в порядке возрастания максимального балла, а для одинаковых максимальных баллов — в алфавитном порядке по фамилии и имени.
6. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх худших студентах по минимальному баллу. В случае, если у нескольких студентов минимальный балл совпадает, то выведите большее число студентов (пока не будут выведены все студенты или не будут полностью исчерпаны студенты с тремя худшими баллами). Вывод надо осуществлять в порядке возрастания минимального балла, а для одинаковых минимальных баллов — в алфавитном порядке по фамилии и имени.
7. С клавиатуры вводится информация об абитуриентах: фамилия, имя, а далее названия предметов и оценки ЕГЭ по ним. Выведите на экран информацию о трёх лучших абитуриентах по максимальному баллу за сумму трёх ЕГЭ. В случае, если у нескольких абитуриентов сумма баллов совпадает, то выведите большее число абитуриентов (пока не будут выведены все абитуриенты или не будут полностью исчерпаны абитуриенты с тремя лучшими баллами). Вывод надо осуществлять в порядке убывания суммы баллов за три ЕГЭ, а для одинаковых сумм — в алфавитном порядке по фамилии и имени.
8. С клавиатуры вводится информация об абитуриентах: фамилия, имя, а далее названия предметов и оценки ЕГЭ по ним. Выведите на экран информацию о трёх худших абитуриентах по максимальному баллу за сумму трёх ЕГЭ. В случае, если у нескольких абитуриентов сумма баллов совпадает, то выведите большее число абитуриентов (пока не будут выведены все абитуриенты или не будут полностью исчерпаны абитуриенты с тремя худшими баллами). Вывод надо осуществлять в порядке возрастания суммы баллов за три ЕГЭ, а для одинаковых сумм — в алфавитном порядке по фамилии и имени.
9. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх студентах с наибольшим количеством пятёрок. В случае, если у нескольких студентов количество пятёрок совпадает, то выведите большее число студентов (пока не будут выведены все студенты или не будут полностью исчерпаны студенты с тремя лучшими показателями). Вывод надо осуществлять в порядке убывания количества пятёрок, а для одинакового количества — в алфавитном порядке по фамилии и имени.
10. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх студентах с наибольшим количеством двоек. В случае, если у нескольких студентов количество двоек совпадает, то выведите большее число студентов (пока не будут выведены все студенты или не будут полностью исчерпаны студенты с тремя худшими показателями). Вывод надо осуществлять в порядке убывания количества двоек, а для одинакового количества — в алфавитном порядке по фамилии и имени.
11. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх студентах с наибольшим разбросом оценок (разница между максимальной и минимальной). В случае, если у нескольких студентов разброс совпадает, то выведите большее число студентов. Вывод в порядке убывания разброса, затем по алфавиту.

12. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх студентах с наименьшим разбросом оценок. В случае, если у нескольких студентов разброс совпадает, то выведите большее число студентов. Вывод в порядке возрастания разброса, затем по алфавиту.
13. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх студентах с наибольшей медианой оценок. В случае совпадения — выведите всех, сортируя по алфавиту.
14. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о трёх студентах с наименьшей медианой оценок. В случае совпадения — выведите всех, сортируя по алфавиту.
15. С клавиатуры вводится информация о студентах: фамилия, имя, курс, оценки. Выведите на экран информацию о лучших студентах по среднему баллу на каждом курсе (по одному с каждого курса). Если на курсе несколько лучших — выведите всех, сортируя по алфавиту.
16. С клавиатуры вводится информация о студентах: фамилия, имя, факультет, оценки. Выведите на экран информацию о лучших студентах по среднему баллу на каждом факультете (по одному с каждого факультета). Если на факультете несколько лучших — выведите всех, сортируя по алфавиту.
17. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о студентах, у которых все оценки — пятёрки. Вывод в алфавитном порядке. Если таких больше трёх — выведите первых трёх по алфавиту.
18. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о студентах, у которых есть хотя бы одна двойка. Отсортируйте по возрастанию среднего балла, затем по алфавиту. Выведите первых трёх.
19. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о студентах, у которых средний балл выше 4.5. Отсортируйте по убыванию среднего балла, затем по алфавиту. Выведите первых трёх.
20. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о студентах, у которых средний балл ниже 3.0. Отсортируйте по возрастанию среднего балла, затем по алфавиту. Выведите первых трёх.
21. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о студентах, имеющих наибольшее количество оценок (например, если у кого-то 5 оценок, а у других — 4). Выведите всех таких, но не более пяти, отсортировав по алфавиту.
22. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о студентах, имеющих наименьшее количество оценок. Выведите всех таких, но не более пяти, отсортировав по алфавиту.
23. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о студентах, у которых мода оценок — 5 (наиболее часто встречающаяся оценка). Отсортируйте по среднему баллу (убывание), затем по алфавиту. Выведите первых трёх.
24. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о студентах, у которых мода оценок — 2. Отсортируйте по среднему баллу (возрастание), затем по алфавиту. Выведите первых трёх.
25. С клавиатуры вводится информация о студентах: фамилия, имя, оценки. Выведите на экран информацию о студентах, у которых стандартное отклонение оценок минимально (наиболее стабильные). Выведите трёх лучших, сортируя по возрастанию отклонения, затем по алфавиту.

#### **Задание №6**

1. По номеру числа Фибоначчи найдите число Фибоначчи (не используйте факты, которые вы не можете доказать самостоятельно)

2. По числу Фибоначчи найдите его номер (не используйте факты, которые вы не можете доказать самостоятельно)
3. По натуральному числу найдите его факториал
4. По факториалу найдите исходное число
5. По данному числу найдите простое число с таким номером (если простые числа нумеровать в порядке возрастания)
6. По простому числу определите его номер в последовательности всех простых чисел, расположенных по возрастанию
7. По данному числу найдите все его простые делители
8. По числу  $n$  найдите  $n$ -ое совершенное число (не используйте факты, которые вы не можете доказать самостоятельно)
9. По совершенному числу найдите его номер в последовательности всех совершенных чисел, расположенных в порядке возрастания (не используйте факты, которые вы не можете доказать самостоятельно)
10. По натуральному числу найдите его двойной факториал
11. По двойному факториалу найдите исходное число
12. Рассмотрим все тройки натуральных чисел, удовлетворяющих уравнению  $a^2 + b^2 = c^2$ . Для данного  $n$  найдите такую тройку чисел  $a, b, c$ , что  $a^2 + b^2 = c^2$ , чтобы  $a + b + c$  было меньше  $n$  и наиболее близко к  $n$ .
13. Рассмотрим все тройки натуральных чисел, удовлетворяющих уравнению  $a^2 + b^2 = c^2$ . Для данного  $n$  найдите такую тройку чисел  $a, b, c$ , что  $a^2 + b^2 = c^2$ , чтобы  $a + b + c$  было больше  $n$  и наиболее близко к  $n$ .
14. По данному натуральному числу  $n$  найдите наименьшее простое число, большее  $n$
15. По данному натуральному числу  $n$  найдите наибольшее простое число, меньшее  $n$
16. По данному натуральному числу  $n$  найдите наименьший факториал, больший  $n$
17. По данному натуральному числу  $n$  найдите наибольший факториал, меньший  $n$
18. По данному натуральному числу  $n$  найдите наименьший двойной факториал, больший  $n$
19. По данному натуральному числу  $n$  найдите наибольший двойной факториал, меньший  $n$
20. По данному натуральному числу  $n$  найдите наименьшее число Фибоначчи, большее  $n$  (не используйте факты, которые вы не можете доказать самостоятельно)
21. По данному натуральному числу  $n$  найдите наибольшее число Фибоначчи, меньшее  $n$  (не используйте факты, которые вы не можете доказать самостоятельно)
22. По данному натуральному числу  $n$  найдите наименьшее совершенное число, большее  $n$  (не используйте факты, которые вы не можете доказать самостоятельно)
23. По данному натуральному числу  $n$  найдите наибольшее совершенное число, меньшее  $n$  (не используйте факты, которые вы не можете доказать самостоятельно)
24. Для данного натурального числа  $n$  найдите такое простое число  $p$ , что входит в разложение на простые множители числа  $n$  наибольшее число раз.
25. Для данного натурального числа  $n$  найдите такое простое число  $p$ , что входит в разложение на простые множители числа  $n$  наименьшее число раз.

## 2.5 Семинар «Поддержка ООП в Kotlin» (4 часа)

Напишите программу, осуществляющую ввод информации о сущностях, описанных в вашем варианте задания, и вывод на экран части из них. Количество вводимых сущностей не ограничено. При реализации обязательно используйте принципы объектно-ориентированного программирования: инкапсуляцию, наследование и полиморфизм, а также следуйте принципам SOLID:

- **Принцип единственной ответственности:** класс не должен одновременно отвечать за хранение/бизнес-логику и за ввод-вывод. Разделяйте эти обязанности.
- **Принцип открытости/закрытости:** ограничивайте возможность некорректного использования классов — как при обычном применении, так и при наследовании и переопределении методов. В то же время обеспечьте гибкость: если наследование и переопределение позволяют адаптировать поведение под конкретные нужды, они должны быть разрешены.
- **Принцип подстановки Барбары Лисков:** если класс А наследует В, то А должен быть корректной реализацией В, и объект типа А должен корректно работать в любом контексте, где ожидается В.
- **Принцип разделения интерфейсов:** проектируйте небольшие, целенаправленные интерфейсы (`interface` в Kotlin), соответствующие конкретным потребностям клиентов. Классы могут реализовывать один или несколько таких интерфейсов. Избегайте «жирных» интерфейсов, заставляющих классы зависеть от методов, которые им не нужны.
- **Принцип инверсии зависимостей:** модули верхнего уровня (конкретные классы) должны зависеть от абстракций (интерфейсов или базовых классов), а не от других конкретных классов. Абстракции не должны зависеть от деталей; детали — от абстракций.

Выберите один из следующих вариантов задания:

1. **Товары интернет-магазина — книги и диски.** Все товары имеют цену. Книги — название, автора, количество страниц. Диски — название, количество треков. Выведите все товары со стоимостью менее 100 рублей.
2. **Преподаватели.** Определяются ФИО. У тех, кто имеет диссертацию, указывается её название; у остальных — стаж работы. Выведите всех преподавателей, чьё ФИО начинается на букву «А».
3. **Телефоны.** Определяются названием модели. Проводные телефоны — типом номеронабирателя (диск или кнопки); беспроводные — дальностью действия радиосигнала. Выведите все телефоны, название которых начинается на «А».
4. **Покатушки.** Определяются названием и расстоянием. Однодневные — плановым временем поездки (в часах). Многодневные — количеством дней и категорией сложности (от 1 до 6). Выведите все покатушки длиной более 100 км.
5. **Музыкальные композиции.** Определяются названием и композитором. Для песен дополнительно указывается автор стихов. Выведите информацию о всех композициях, у которых композитор начинается на «А».
6. **Олимпиады.** Определяются названием. Если олимпиада участвует в программе приёма в вузы — указывается уровень (1–3). Если это этап Всероссийской олимпиады — название этапа (школьный, окружной, региональный, всероссийский). В остальных случаях — размер призового фонда. Выведите все олимпиады, название которых начинается на «А».
7. **Проездные билеты.** Определяются стоимостью. Билет на количество поездок — количеством поездок. Билет на неограниченное количество поездок — сроком действия (1 день, 5, 10, 15 дней, месяц, 3 месяца, 6 месяцев, год). Выведите билеты стоимостью менее 300 рублей.
8. **Студенты.** Определяются ФИО. У студентов без автомата — балл за экзамен (2–5). У студентов с автоматом — основание (олимпиада или контрольные работы); если основание — контрольные, то также указывается средний балл за них. Выведите информацию о студентах с фамилией, начинающейся на буквы от «А» до «К».

9. **Сотовые телефоны.** Определяются названием. У смартфонов — операционная система; у остальных — наличие браузера. Выведите телефоны, название которых содержит слово «Nokia».
10. **Куртки.** Определяются названием модели и наличием капюшона. Мембранные куртки — степенью водонепроницаемости (в мм вод. ст.); остальные — наличием пропитки. Выведите информацию обо всех куртках с капюшоном.
11. **Жёсткие диски.** Определяются названием и ёмкостью. Внешние — наличием системы защиты от падения; внутренние — размером (2.5" или 3.5"). Выведите диски ёмкостью более 200 ГБ.
12. **Велосипеды.** Определяются названием модели. Горные — количеством скоростей; BMX — типом конструкции (фривил, кассетная, фрикостер). Выведите информацию обо всех велосипедах, в названии которых есть «Norco».
13. **Электронные книги.** Определяются названием и размером экрана. У E-Ink — поколение дисплея (Pearl, Vizplex); у LCD — количество поддерживаемых цветов. Выведите книги с экраном не менее 7 дюймов.
14. **GPS-навигаторы.** Определяются названием и диагональю экрана. Переносные — наличием велокрепления; автомобильные — поддержкой отображения пробок и наличием радар-детектора. Выведите GPS с экраном менее 7 дюймов.
15. **Пылесосы.** Определяются названием модели. Обычные — мощностью; роботы-пылесосы — площадью убираемого помещения и количеством виртуальных стен. Выведите пылесосы, в названии которых есть «Indesit».
16. **Туры.** Определяются названием. Пляжные — типом пляжа (галечный, песок); экскурсионные — количеством экскурсий. Выведите туры, содержащие слово «Египет».
17. **Языки программирования.** Определяются названием. Алгоритмические языки — поддержкой ООП (отсутствует, на классах, прототипная); остальные — типом (функциональный, логический, императивный и т.п.). Выведите языки, название которых начинается на «А».
18. **Контрагенты.** Определяются названием. ИП — наличием банковского счёта; юридические лица — формой организации (ООО, ОАО, ЗАО). Выведите контрагентов, название которых начинается на «А».
19. **Банковские счета.** Определяются номером. Текущие — платой за обслуживание; сберегательные — процентной ставкой и наличием капитализации. Выведите счета, номер которых начинается с «408178».
20. **Автомобильные дороги.** Определяются названием и длиной (в км). Бесплатные — статусом автомагистрали (да/нет); платные — стоимостью за км. Выведите дороги длиной менее 100 км.
21. **Офисные здания.** Определяются адресом. При наличии стоянки — количеством машиномест и стоимостью аренды в месяц. Выведите здания, в адресе которых есть слово «Тверская».
22. **Товары интернет-магазина — GPS-навигаторы и карты.** Все товары — ценой и названием. GPS — назначением (ручной, автомобильный) и возможностью загрузки карт; карты — размером (в МБ). Выведите товары стоимостью менее 4000 рублей.
23. **Товары интернет-магазина — чай и кофе.** Все товары — ценой, названием и весом. Кофе — типом (растворимый, молотый, в зёрнах); чай — типом (чёрный, зелёный). Выведите товары весом менее 150 г.
24. **Объекты в коттеджном посёлке — участки и дома.** Участки — площадью, стоимостью, наличием подряда. Дома — этажностью, площадью, стоимостью. Выведите объекты стоимостью менее 1 000 000 рублей.
25. **Вопросы теста.** Общие поля — формулировка и баллы за правильный ответ. Вопросы с вариантами — 4 варианта и номер правильного; остальные — формулировка правильного ответа. Выведите вопросы, оцениваемые в 10 баллов и выше.

26. **Слова.** Определяются самим словом. Существительные — родом; глаголы — спряжением. Выведите слова, начинающиеся на «А».
27. **Операционные системы.** Определяются названием. На базе Linux — названием менеджера пакетов; остальные — стоимостью лицензии. Выведите ОС, название которых начинается на «А».
28. **Рюкзаки.** Определяются названием модели и ёмкостью. Городские — наличием «вентилируемой спины»; походные — количеством отделений и креплением для трекинговых палок. Выведите рюкзаки, в названии которых есть «Trek».
29. **Автостоянки.** Определяются названием и количеством машиномест. Крытые — количеством этажей; открытые — наличием охраны. Выведите стоянки с более чем 20 местами.
30. **Политические партии.** Определяются названием. Финансируемые из бюджета — размером ассигнований; остальные — количеством депутатов в представительных органах. Выведите партии, название которых начинается на буквы от «А» до «К».

### **3 Семинар для рубежного контроля**

На семинаре будет предложено решить на листочке одно из заданий, предложенных к выполнению в течение семестра.

### **4 Разработка на платформе Android**

## 4.1 Семинар «Простейшие программы для Android» (2 часа)

Разработайте программу, работающую под управлением Android с использованием Jetpack Compose. Проверьте, что программа корректно работает с различными размерами экрана, а также при повороте экрана. Необходимо проверять корректность исходных данных и сообщать о проблемах.

### Варианты для группы 411

- 1 Программа-календарь (по году, месяцу, числу – день недели) с удобным вводом (с выпадающим списком). В работе запрещено использовать встроенные классы для даты и ввода даты.
- 2 Программа нахождения решения неравенства  $ax + b \text{ знак}$  (удобный ввод с выпадающим списком знака неравенства).
- 3 Программа нахождения решения неравенства  $ax^2 + bx + c \text{ знак}$  (удобный ввод с выпадающим списком знака неравенства).
- 4 Программа учета расходов и доходов по месяцам: в элементы управления вводятся доходы за январь-декабрь и расходы за январь-декабрь, программа выводит прибыль за январь-декабрь, общую сумму расходов, доходов и прибыли за год. Интерфейс должен напоминать таблицу.
- 5 Программа поиска обратной матрицы для матрицы  $3 \times 3$ . Запрещено использовать специальные библиотеки для матриц.
- 6 Программа нахождения центра и радиуса окружности по координатам трех точек, принадлежащих ей.
- 7 Напишите игру «Пятнашки» (не следует делать идеальный интерфейс)
- 8 Программа решения системы уравнений

$$\begin{cases} ax + by + c = 0 \\ dx + ey + f = 0 \end{cases}$$

где все представленные числа комплексные. Программа должна находить ответ в случае, если решение единственно, в противном случае выдавать диагностическое сообщение о невозможности решения. Использовать специальные классы для комплексных чисел, если они не реализованы вами, запрещено

- 9 Программа перевода даты (год, месяц, число) из григорианского календаря в юлианский и обратно с удобным вводом (с выпадающим списком). В работе запрещено использовать встроенные классы для даты и ввода даты.
- 10 Программа нахождения площади меньшей из частей круга, порождаемых пересечением круга и прямой, по координатам центра и радиусу круга и по координатам двух точек прямой.
- 11 Программа деления двух многочленов: не более, чем четвертой степени, на не более, чем второй.
- 12 Программа-игра чет-нечет, где человек задумывает, а программа отгадывает четность/нечетность загаданных чисел. Программа должна анализировать загадываемые человеком числа с целью предсказания, выявляя закономерности длиной от 1 до 10 чисел: в случае если человек регулярно загадывает одинаковую последовательность чисел такой длины, то программа с некоторого момента должна их угадывать всегда. Если человек чаще всего после некой последовательности длины от 1 до 10 чисел загадывает чет (или нечет), то программа данное явление также должна выявлять.

Не следует придумывать слишком сложный алгоритм. Это не ИИ (в современном смысле слова).

- 13 Программа решения уравнения  $ax^2 + bx + c = 0$ , где все представленные числа – комплексные. Запрещено использовать классы для представления комплексных чисел, если они не реализованы вами.

- 14 Программа-калькулятор для матриц  $2 \times 2$ . Пользователю доступно задание матриц  $2 \times 2$  и кнопки  $+$ ,  $-$ ,  $*$ ,  $/$ ,  $=$ ,  $C$
- 15 Программа перевода чисел в/из римской системы счисления
- 16 Программа перевода числа из одной системы счисления в другую (от 2 до 36) с удобным вводом: пользователь указывает исходную и целевую системы через выпадающие списки и вводит число. Запрещено использовать функции библиотеки для перевода из одной системы счисления в другую.
- 17 Программа нахождения точки пересечения двух отрезков на плоскости по координатам их концов. В случае отсутствия пересечения — вывод соответствующего сообщения.
- 18 Программа шифрования и дешифрования текста шифром Цезаря с возможностью выбора языка (русский/английский) и сдвига через интерфейс. Программа должна корректно обрабатывать регистр и неалфавитные символы.
- 19 Пользователь выбирает тип операции ( $+$ ,  $-$ ,  $\times$ ,  $\div$ ), диапазон чисел и количество вопросов. Программа генерирует примеры, принимает ответы, ведёт счёт правильных/неправильных и в конце выводит статистику. Деление должно быть только нацело (делимое кратно делителю).
- 20 **Программа нахождения координат точки пересечения медиан треугольника (центроида).** Пользователь вводит координаты трёх вершин треугольника. Программа вычисляет и выводит координаты центроида как среднее арифметическое координат вершин.
- 21 **Программа нахождения длины общей хорды двух пересекающихся окружностей.** Пользователь вводит центры и радиусы двух окружностей, которые гарантированно пересекаются в двух точках. Программа вычисляет и выводит длину их общей хорды (через геометрические соотношения в треугольниках, образованных центрами и точками пересечения).
- 22 **Программа нахождения координат центра и радиуса вписанной окружности треугольника.** Пользователь вводит координаты трёх вершин невырожденного треугольника. Программа вычисляет и выводит координаты центра вписанной окружности и её радиус.
- 23 **Программа нахождения расстояния между двумя непересекающимися отрезками на плоскости.** Пользователь вводит координаты концов двух отрезков. Программа определяет, пересекаются ли они; если да — выводит 0; если нет — вычисляет и выводит минимальное расстояние между ними (минимальное расстояние между точками, принадлежащими разным отрезкам).
- 24 **Программа нахождения уравнения общей касательной к двум окружностям.** Пользователь вводит центры и радиусы двух окружностей. Программа выводит коэффициенты уравнения прямой  $Ax + By + C = 0$ , являющейся одной из таких касательных (любой) или сообщает об отсутствии таковой.
- 25 **Программа нахождения площади пересечения двух прямоугольников со сторонами, параллельными осям координат.** Пользователь вводит координаты противоположных углов двух таких прямоугольников. Программа вычисляет площадь их пересечения (0, если не пересекаются).
- 26 **Программа нахождения координат точки пересечения высот треугольника.** Пользователь вводит координаты трёх вершин невырожденного треугольника. Программа вычисляет и выводит координаты точки пересечения трёх высот.

#### Варианты для группы 412

1. Программа решения квадратного уравнения
2. Программа решения неравенства вида  $ax + b > 0$
3. Программа решения неравенства вида  $ax + b < 0$
4. Программа решения неравенства вида  $ax + b \geq 0$
5. Программа решения неравенства вида  $ax + b \leq 0$

6. Программа поиска дня недели по числу и месяцу в текущем году
7. Программа перевода числа из 10-ой в 16-ую, 8-ую и 2-ую систем.
8. Программа поиска времени, когда окончится интервал. Дано: часы и минуты начала интервала и количество минут, сколько он идет. Результат: часы и минуты окончания интервала.
9. Программа поиска обратной матрицы для матрицы  $3 \times 3$ .
10. Программа поиска длины интервала. Дано: часы и минуты начала интервала и часы и минуты конца интервала. Результат: количество минут в интервале.
11. Программа умножения и деления двух комплексных чисел.
12. Программа нахождения площади треугольника по координатам вершин.
13. Программа нахождения углов треугольника по координатам вершин (проще всего это сделать по теореме косинусов).
14. Программа перевода числа из 16-ой, 8-ой и 2-ой системы в 10-ую систему счисления.
15. Программа нахождения количества денег на вкладе после окончания его срока по начальному взносу, проценту и срока в годах.
16. Программа нахождения степени комплексного числа. Исходные данные: действительная, мнимая часть числа и степень. Результат: действительная и мнимая часть результата.
17. Программа умножения и деления чисел, представленных в виде обыкновенных дробей (состоящих из целой части, числителя и знаменателя). Не забудьте выполнить сокращение дроби и приведение ее к правильному виду.
18. Программа сложения и вычитания чисел, представленных в виде обыкновенных дробей (состоящих из целой части, числителя и знаменателя). Не забудьте выполнить сокращение дроби и приведение ее к правильному виду.
19. Программа определения по дате (число и месяц) знака зодиака.
20. Программа определения по обыкновенной дроби (числителю и знаменателю) периода десятичной дроби.
21. Программа перевода комплексного числа из обычной формы в тригонометрическую и наоборот.
22. Программа-игра Баше. При реализации этого задания не требуется ничего рисовать, вся информация вводится и выводится в виде чисел в обычные элементы управления.
23. Программа разложения числа на простые множители.
24. Программа нахождения наибольшего общего делителя и наименьшего общего кратного двух натуральных чисел.
25. Программа-тест по предмету «Разработка мобильных приложений». Создайте программу-тест из 10 вопросов с выбором вариантов ответов и показом результатов прохождения теста.
26. Программа генерации всех перестановок заданного набора различных символов (например, букв или цифр). Исходные данные: строка из  $k$  различных символов ( $1 \leq k \leq 8$ ). Результат: список всех возможных перестановок в лексикографическом порядке.
27. Программа нахождения всех целых решений (пар  $(x, y)$ ) линейного диофантова уравнения вида  $ax + by = c$ , где  $a, b, c$  — целые числа, в заданном диапазоне (например,  $-100 \leq x, y \leq 100$ ). Исходные данные: коэффициенты  $a, b, c$  и границы диапазона поиска. Результат: список всех целочисленных пар  $(x, y)$ , удовлетворяющих уравнению в указанном диапазоне, либо сообщение об отсутствии решений.
28. Программа решения системы двух линейных уравнений с двумя неизвестными методом Крамера. Дано: коэффициенты уравнений  $a_1x + b_1y = c_1$ ,  $a_2x + b_2y = c_2$ . Результат: значения  $x$  и  $y$ , либо сообщение о том, что система не имеет единственного решения (определитель равен нулю).

## 4.2 Семинар «MVVM» (2 часа)

Реализуйте программу, созданную в ходе работы над прошлым семинаром, с использованием паттерна MVVM. Обеспечьте автоматическое UNIT-тестирование модели в соответствии с принципами UNIT-тестирования.

### 4.3 Семинар «Работа со списками» (4 часа)

В данной работе вам необходимо взять код с семинара «Объектно-ориентированное программирование» и разделить его на три части:

- модуль **core** (наведите курсор мыши на проект, нажмите правую кнопку мыши и создайте модуль Java/Kotlin library);
- модуль **console** (консольное приложение, которое работает как раньше и использует модуль core);
- модуль **app** (он создается мастером по умолчанию; Android-приложение, которое использует модуль core).

В последних двух модулях подключение core делается посредством строки:

```
implementation(project(":core"))
```

Реализуйте функциональность консольного приложения в Android-приложении, не забыв выделить отдельно View, ViewModel, а также Model, которая преимущественно реализуется модулем core. Используйте Navigation для переходов между экранами.

Дополните Android-приложение полной CRUD-функциональностью (создание, просмотр, обновление и удаление сущностей согласно вашему варианту).

#### 4.4 Семинары «Использование библиотек для поддержки Dependency Injection и Clean Architecture» (2 часа)

Используйте Hilt для внедрения Dependency Injection в проект, созданный на прошлом семинаре. Приведите проект в соответствие принципам Clean Architecture (выделить View, ViewModel, UseCases, Model, Repositories, DataSources, DI).

Для хранения информации в Android-приложении (и только в нем) используйте ROOM (соответственно у вас будет три репозитория – тот, что был реализован в консоли; ROOM-репозиторий и Mock).

#### **4.5 Семинар «Работа с внешними сервисами RestAPI» (2 часа)**

Реализуйте клиент для сервиса, что вы реализовали в курсе «Программирование на Java» (минимально работоспособный, можно без архитектуры).

## 5 Семинар для подготовки к зачёту

На семинаре будет предложено решить на компьютере ВУЗа одно из заданий, предложенных к выполнению в течение семестра (упрощённое, чтобы можно было уложиться в 60 минут).

## 6 Проекты как замена заданиям семинаров

Студент по согласованию с преподавателем может реализовывать мобильное приложение, являющееся крупным проектом (его объём должен быть не меньше суммарного объёма заданий семинаров).

Проект может реализовываться на Android (Kotlin), iOS (Swift) или Flutter (Dart).

В ходе реализации проекта нужно использовать следующие принципы: покрытие UNIT- и UI-тестами, принципы Clean Architecture, MVVM, DI с использованием стандартных библиотек, SOLID, функциональный подход в программировании.

Используйте стандартные средства навигации (Navigation) или их аналоги на других платформах.

Код должен соответствовать стандартам кодирования, принятым в крупных организациях, занимающихся разработкой программного обеспечения.

## 7 Зачёт

Зачёт – накопительная оценка, состоящая из оценок за каждое задание (или альтернативный проект), двух итоговых заданий с программированием прямо на парах, а также оценки за теорию (как минимум два письменных опроса на лекциях).

Оценка суммируется следующим образом (каждая базовая оценка в 100-бальной шкале):

$$0,2 \times \text{средняя оценка за практики} + \\ + 0,5 \times \text{средняя оценка за письменные опросы} + \\ + 0,3 \times \text{средняя оценка за практические офлайн-задания}$$

Таблица перевода из 100-бальной шкалы в 4-бальную:

оценка в 100-бальной шкале	оценка в 4-бальной шкале
0,00-50,00	2
50,00-67,99	3
68,00-83,99	4
84,00-100,00	5

## 8 Курсовая работа

Выполнение курсовой работы состоит из следующих этапов (для типичной – разработка продукта):

1. выбор/согласование темы курсовой работы (первые две недели);
2. формулирование индивидуального задания преподавателем (третья неделя)
3. разработка введения в курсовую работу (мотивировки, цели, задачи) до 5 недели;
4. формулирование списка требований (спецификации) до 8 недели;
5. проектирование (выбор инструментальных средств, диаграммы UML, дизайн и так далее) до 11 недели;
6. реализация (описание этапов реализации, использованных паттернов, проблем); собственно программирование будет осуществляться с середины семестра, объём программистской работы должен быть примерно на один месяц (с учетом параллельной учебы и выполнения заданий по другим предметам);
7. тестирование (условно разделено с реализацией: UNIT- и UI-тестирование); до 15 недели
8. формулирование заключения;

9. оформление курсовой работы в соответствии с ГОСТ 7.32-2017 (списка литературы в соответствии с ГОСТ Р 7.0.100-2018) до 16 недели.

Выполнение курсовой работы состоит из следующих этапов (для исследовательской):

1. выбор/согласование темы курсовой работы (первые две недели);
2. формулирование индивидуального задания преподавателем (третья неделя);
3. формулирование гипотезы;
4. разработка введения в курсовую работу (мотивировки, объект, предмет, методы);
5. исследование литературы (теоретическая часть курсового проекта);
6. разработка методики (дизайна) исследования;
7. собственно исследование;
8. разработка выводов по результатам исследования;
9. формулирование заключения;
10. оформление курсовой работы в соответствии с ГОСТ 7.32-2017 (списка литературы в соответствии с ГОСТ Р 7.0.100-2018).

Оценка курсовой работы осуществляется по следующим критериям:

1. выполнение работы в срок и по этапам;
2. качество реализованного продукта/исследования;
3. содержательно корректный текст пояснительной записки;
4. правильность оформления пояснительной записки;
5. качество защиты курсовой работы.

## 9 Список литературы

1. Официальная документация по языку Kotlin: <https://kotlinlang.org/docs/>
2. Официальная документация по платформе Android: <https://developer.android.com/docs>
3. Быстрое введение в Kotlin от авторов языка: <https://stepik.org/course/4222>