

ПЕТРОЗАВОДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНСТИТУТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

09.03.04 – Программная инженерия

Практическое задание по дисциплине «Верификация программного обеспечения»

МОБИЛЬНАЯ ИГРЫ «МОЗГОВИК»

Выполнили:

студент 4 курса группы 22407

Курочкин Дмитрий Александрович \_\_\_\_\_  
*подпись*

]

Руководитель:

К.А. Кулаков, к.ф.-м.н.

\_\_\_\_\_  
*подпись*

# Оглавление

<b>1</b>	<b>Объект тестирования</b>	<b>4</b>
<b>2</b>	<b>Стратегия тестирования</b>	<b>6</b>
2.1	Архитектура приложения . . . . .	6
2.1.1	Пользовательский интерфейс . . . . .	6
2.1.2	Подсистема выбора мини-игры . . . . .	7
2.1.3	Подсистема генерации мини-игры . . . . .	7
2.1.4	Подсистема прохождения мини-игры . . . . .	8
2.2	Описание стратегии блочного тестирования . . . . .	9
2.2.1	Метод проведения . . . . .	9
2.2.2	Используемые окружение и инструменты . . . . .	9
2.2.3	Способ оценивания результатов . . . . .	9
2.3	Описание стратегии интеграционного тестирования . . . . .	10
2.3.1	Метод проведения . . . . .	10
2.3.2	Используемые окружение и инструменты . . . . .	10
2.3.3	Способ оценивания результатов . . . . .	10
2.4	Описание стратегии аттестационного тестирования . . . . .	11
2.4.1	Метод проведения . . . . .	11
2.4.2	Используемые окружение и инструменты . . . . .	11
2.4.3	Способ оценивания результатов . . . . .	11
2.5	Описание стратегии специальных видов тестов . . . . .	11
2.5.1	Метод проведения . . . . .	11
2.5.2	Используемые окружение и инструменты . . . . .	12
2.5.3	Способ оценивания результатов . . . . .	12
2.6	Условия начала, окончания и перехода между этапами тестирования . . . . .	12
2.7	Условия возобновления и приостановки выполнения тестов. . . . .	13

<b>3</b>	<b>Детальный план тестов</b>	<b>14</b>
3.1	Блочное тестирование . . . . .	14
3.1.1	Подсистема выбора мини-игры . . . . .	14
3.1.2	Подсистема генерации мини-игры . . . . .	17
3.1.3	Подсистема прохождения мини-игры . . . . .	27
3.2	Интеграционное тестирование . . . . .	41
3.3	Аттестационное тестирование . . . . .	45
3.4	Нагрузочное тестирование . . . . .	53
<b>4</b>	<b>Примеры тестов</b>	<b>55</b>
4.1	Блочный тест Б23 . . . . .	55
4.2	Интеграционный тест И1 . . . . .	56
<b>5</b>	<b>Покрытие кода тестами</b>	<b>58</b>
<b>6</b>	<b>Журнал тестирования</b>	<b>59</b>
<b>7</b>	<b>Журнал ошибок</b>	<b>62</b>
<b>8</b>	<b>Результаты тестирования</b>	<b>64</b>

# Глава 1

## Объект тестирования

Объектом тестирования является мобильная игра под систему android под названием "Мозговик". Приложение представляет собой набор нескольких мини-игр, среди которых: решение уравнения, «перевёртыши». В каждой игре можно выбрать уровень сложности, в игре фиксируется рекорд игрока по времени решения мини-игры.

Пользователь может:

1. Выбрать мини-игру.
2. Выбрать уровень сложности в мини-игре.
3. Посмотреть рекорд в игре на каждом уровне сложности.
4. Пройти мини-игру.

Участие функциональностей в аттестационных тестах:

1. Выбор мини-игры:
  - (a) Пользователь может выбрать каждую из предложенных мини-игр.
  - (b) Пользователь может посмотреть описание каждой из предложенных мини-игр.
2. Выбор сложности:
  - (a) Уровень сложности меняет время, данное игроку на прохождение уровня.
3. Просмотр рекордов:
  - (a) Для каждой мини-игры на каждом уровне сложности свой рекорд.
  - (b) Показывается последний лучший рекорд.

(с) Фиксируется рекорд только мини-игр, завершённых победой.

#### 4. Прохождение мини-игры

(а) Мини-игра имеет два состояния завершения: победа, проигрыш.

(b) Если мини-игра была пройдена за отведённое время, то она завершается победой.

(с) Если игра не была пройдена за отведённое время, то она завершается проигрышем.

# Глава 2

## Стратегия тестирования

### 2.1 Архитектура приложения



Рис. 2.1: Архитектура приложения]

#### 2.1.1 Пользовательский интерфейс

Графический пользовательский интерфейс в виде экранов. Все экраны реализованы в формате xml.

Пункты аттестационных тестов: 1, 2, 3, 4.

Файлы:

1. activity\_achivement.xml
2. activity\_analiticks.xml

3. activity\_main.xml
4. activity\_pecsesso.xml
5. activity\_training.xml
6. game\_info\_layout.xml
7. line\_of\_number.xml

### 2.1.2 Подсистема выбора мини-игры

Перечень мини-игр с информацией о них (название и описание), о рекорде и выбор уровня сложности.

Пункты аттестационных тестов: 1, 3.

Файлы:

1. TrainingActivity.kt

Классы:

1. TrainingActivity

(a) onStartGameButtonClick(view: View)

Включает выбранную игру с выбранным уровнем сложности

### 2.1.3 Подсистема генерации мини-игры

Генерация выбранной мини-игры с учётом выбранной сложности со случайными условиями.

Пункты аттестационных тестов: 2

Файлы:

1. LineOfNumber.kt
2. Pecsesso.kt

Классы:

1. LineOfNumber

(a) createGame(diff: Int)

Генерирует игру «Числовой ряд» со сложностью diff

(b) `sing(): List<String>`

Возвращает последовательность арифметических операций для создания примера

(c) `number(signs: List<String>): List<Int>`

На вход получает последовательность символов, на выходе даёт готовый пример

(d) `hideNumber(layoutNumbers: List<EditText>, diff: Int)`

На вход получает готовый пример и уровень сложности. Скрывает части примера, в соответствии со сложностью в произвольном порядке.

## 2. Pecsesso

(a) `newGame(view: View)` На вход получает информацию с экрана. Запускает процесс создания новой игры.

(b) `startGame(diff: Int)` На вход получает уровень сложности, создаёт условия новой игры.

(c) `potasovka()` «Размешивает» карты на столе.

### 2.1.4 Подсистема прохождения мини-игры

Возможность прохождения мини-игры, ведение отсчёта времени и определения состояний победы/поражения.

Пункты аттестационных тестов: 4

Файлы:

1. `LineOfNumber.kt`

2. `Pecsesso.kt`

Классы:

1. `LineOfNumber`

(a) `startTimer()` Запускает таймер на прохождение мини-игры.

(b) `lineOfNumberCheck(view: View)`

На вход получает информацию с экрана. Проверяет правильность решения примера

(c) `endGame()` Проверяет истечение времени таймера, в случае исхода времени - заканчивает игру поражением.



## 2. Pecesso

- (a) `onCardClick(view: View)` На вход получает информацию с экрана. Проверяет наличие пары у перевёрнутой карты и активирует победу в случае обнаружения всех пар.
- (b) `endGame()` Проверяет истечение времени таймера, в случае исхода времени - заканчивает игру поражением.
- (c) `areDrawablesIdentical(drawableA: Drawable, drawableB: Drawable): Boolean` Проверяет идентичность перевёрнутых карт.
- (d) `startTimer()` Запускает таймер на прохождение мини-игры.

## 2.2 Описание стратегии блочного тестирования

### 2.2.1 Метод проведения

- Черный ящик (Black Box):  
Тестирование, ориентированное на функциональность блока без знания внутренней реализации. Это позволяет проверить соответствие требованиям.
- Белый ящик (White Box):  
Исследование внутренней структуры блока с целью тестирования логики и структуры кода. Это может включать в себя тестирование путей выполнения, ветвей и условий.

### 2.2.2 Используемые окружение и инструменты

- Интегрированные среды разработки (IDE):  
Использование IDE для проведения тестов, анализа кода и отладки.
- Автоматизированные инструменты тестирования:  
Применение инструментов, таких как JUnit для автоматизации тестовых сценариев.

### 2.2.3 Способ оценивания результатов

- Проверка соответствия требованиям:  
Оценка того, соответствует ли блок тем требованиям, которые были определены на этапе проектирования.

- **Покрытие кода:**

Измерение процента кода, который был протестирован в ходе блочного тестирования.

## **2.3 Описание стратегии интеграционного тестирования**

### **2.3.1 Метод проведения**

- **Постепенная интеграция (Incremental Integration):**

Последовательное добавление и тестирование новых компонентов в систему.

- **Вертикальная интеграция (Top-Down):**

Тестирование отдельных верхних уровней системы вместе с подчиненными компонентами.

- **Горизонтальная интеграция (Bottom-Up):**

Тестирование низкоуровневых компонентов, постепенно объединяя их в более высокоуровневые модули.

### **2.3.2 Используемые окружение и инструменты**

- **Интеграционные фреймворки:**

Использование инструментов и фреймворков, такой, как JUnit и Robolectric, для автоматизации интеграционных тестов.

- **Средства мониторинга:**

Применение средств для отслеживания и анализа взаимодействия компонентов в реальном времени.

### **2.3.3 Способ оценивания результатов**

- **Выполнение взаимодействия частей системы согласно требованиям:**

Оценка взаимозависимости процесса обмена информацией при интегрировании подсистем.

- **Соответствие функциональным требованиям:**

Проверка того, что интегрированные компоненты взаимодействуют в соответствии с функциональными требованиями.

## 2.4 Описание стратегии аттестационного тестирования

### 2.4.1 Метод проведения

- Функциональное тестирование:  
Проверка соответствия функциональных требований системы.
- Нефункциональное тестирование:  
Оценка характеристик, таких как производительность, безопасность, масштабируемость.

### 2.4.2 Используемые окружение и инструменты

- Среды воспроизведения:  
Тестирование работы приложения на эмуляторе реальных телефонов.
- Инструменты тестирования производительности:  
Использование инструментов среды Android studio, для оценки производительности системы.

### 2.4.3 Способ оценивания результатов

- Соответствие требованиям и стандартам:  
Проверка того, что система соответствует всем установленным требованиям и стандартам.
- Полнота тестового покрытия:  
Оценка того, насколько хорошо тестовые сценарии покрывают функциональные и нефункциональные аспекты системы.

## 2.5 Описание стратегии специальных видов тестов

### 2.5.1 Метод проведения

- Тестирование максимальной нагрузки (Load Testing):  
Оценка производительности системы при максимальной предполагаемой нагрузке.
- Тестирование устойчивости (Stress Testing):  
Проверка устойчивости системы при экстремальных условиях и перегрузке.

- Тестирование масштабируемости (Scalability Testing):

Исследование способности системы эффективно масштабироваться при увеличении нагрузки.

## **2.5.2 Используемое окружение и инструменты**

- Инструменты IDE Android Studio для оценки производительности системы.

## **2.5.3 Способ оценивания результатов**

- Время отклика и задержки:

Измерение времени, требуемого для обработки запросов при различных уровнях нагрузки.

## **2.6 Условия начала, окончания и перехода между этапами тестирования**

Условия начала тестирования:

- Завершение разработки
- Подготовка тестовых артефактов
- Назначение ресурсов
- Определение критериев приемлемости

Условия завершения тестирования:

- Выполнение всех тестов
- Анализ результатов
- Удовлетворение критериев качества
- Согласование с заказчиком

Условия переходов между этапами:

Переход от модульного к интеграционному тестированию происходит при выполнении минимум 90% всех модульных тестов

## 2.7 Условия возобновления и приостановки выполнения тестов.

Условия возобновления:

- Исправление критических дефектов
- Изменение в коде
- Обновление тестовых данных
- Актуализация тестовых сценариев

Условия приостановки:

- Выполнение критических задач
- Нехватка ресурсов
- Неопределенность в требованиях
- Подготовка окружения
- Ожидание новой функциональности

## Глава 3

# Детальный план тестов

### 3.1 Блочное тестирование

#### 3.1.1 Подсистема выбора мини-игры

Класс «TrainingActivity»

Перед тестированием:

- Подготовить моки для SharedPreferences, SharedPreferences.Editor, Spinner, View.
- Установить diff значение 1 или 2.

Номер	Б1
Цель	Проверка работы при инициализированных данных при выборе игры «Ряд чисел»
Тип	Позитивный
Объект тестирования	функция «onStartGameButtonClick»
Входные данные	View со значением spinner_game.text «Числовой ряд»
Косвенные входные данные	<ul style="list-style-type: none"> <li>• diff = 1</li> <li>• SharedPreferences, SharedPreferences.Editor, Spinner, View инициализированы</li> </ul>
Ожидаемый результат	открытие активности экрана "LineOfNumber" со сложностью diff

Номер	Б2
Цель	Проверка работы при инициализированных данных при выборе игры «Перевертыши»
Тип	Позитивный
Объект тестирования	Функция onStartGameButtonClick
Входные данные	View со значением spinner_game.text «Перевертыши»
Косвенные входные данные	<ul style="list-style-type: none"> <li>• diff = 1</li> <li>• SharedPreferences, SharedPreferences.Editor, Spinner, View инициализированы</li> </ul>
Ожидаемый результат	Открытие активности экрана «Peccesso» со сложностью diff

Номер	Б3
Цель	Проверка работы при неинициализированных данных (игра не выбрана)
Тип	Негативный
Объект тестирования	Функция <code>onStartGameButtonClick</code>
Входные данные	View со значением <code>spinner_game.text</code> «»
Косвенные входные данные	<ul style="list-style-type: none"> <li>• <code>diff = 1</code></li> <li>• <code>SharedPreferences</code>, <code>SharedPreferences.Editor</code>, <code>Spinner</code>, <code>View</code> инициализированы</li> </ul>
Ожидаемый результат	Отображение Toast сообщения с текстом «Игра не выбрана»

Номер	Б4
Цель	Проверка обработки ошибки при сохранении данных в <code>SharedPreferences</code>
Тип	Негативный
Объект тестирования	Функция <code>onStartGameButtonClick</code>
Входные данные	View со значением <code>spinner_game.text</code> «Числовой ряд»
Косвенные входные данные	<ul style="list-style-type: none"> <li>• <code>diff = 1</code></li> <li>• <code>SharedPreferences.Editor</code> выбрасывает исключение при вызове <code>commit()</code></li> <li>• <code>SharedPreferences</code>, <code>Spinner</code>, <code>View</code> инициализированы</li> </ul>
Ожидаемый результат	Не открытие активности, отображение Toast сообщения с текстом «Игра не выбрана»



### 3.1.2 Подсистема генерации мини-игры

Класс «LineOfNumber»

Предварительные действия не требуются

Номер	Б5
Цель	Проверка работы при инициализированных данных с diff 1
Тип	Позитивный
Объект тестирования	Функция <code>createGame</code>
Входные данные	–
Косвенные входные данные	<ul style="list-style-type: none"> <li>• Элементы массивов <code>layoutSings</code> и <code>layoutNumbers</code> инициализированы</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• <code>diff = 2</code></li> <li>• Установка значений текстовых полей <code>layoutSings</code> и <code>layoutNumbers</code> в соответствии с результатами <code>sing()</code> и <code>number()</code></li> <li>• Установка текста в элемент <code>R.id.answer</code> в значение <code>numbers[4]</code></li> <li>• Вызов функции <code>hideNumber</code> с параметрами <code>layoutNumbers</code> и <code>diff</code></li> </ul>

Номер	Б6
Цель	Проверка работы при инициализированных данных с diff 2
Тип	Позитивный
Объект тестирования	Функция <code>createGame</code>
Входные данные	—
Косвенные входные данные	<ul style="list-style-type: none"> <li>• <code>diff = 2</code></li> <li>• <code>end = false</code></li> <li>• Результаты функций <code>sing()</code> и <code>number()</code> вызваны</li> <li>• Элементы массивов <code>layoutSings</code> и <code>layoutNumbers</code> инициализированы</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Установка значений текстовых полей <code>layoutSings</code> и <code>layoutNumbers</code> в соответствии с результатами <code>sing()</code> и <code>number()</code></li> <li>• Установка текста в элемент <code>R.id.answer</code> в значение <code>numbers[4]</code></li> <li>• Вызов функции <code>hideNumber</code> с параметрами <code>layoutNumbers</code> и <code>diff</code></li> </ul>

Номер	Б7
Цель	Проверка работы при инициализированных данных
Тип	Позитивный
Объект тестирования	Функция <code>sing</code>
Входные данные	Нет входных данных (функция не имеет параметров)
Косвенные входные данные	—
Ожидаемый результат	Создание списка строк длиной 3 (индексы 0, 1, 2) с операторами, выбранными из <code>sings</code>

Номер	Б8
Цель	Проверка уникальности полученных значений
Тип	Позитивный
Объект тестирования	Функция <code>sign</code>
Входные данные	Нет входных данных (функция не имеет параметров)
Косвенные входные данные	—
Ожидаемый результат	Два списка, сгенерированных функцией <code>sign</code> отличаются друг от друга.

Номер	Б9
Цель	Проверка работы при инициализированных данных
Тип	Позитивный
Объект тестирования	Функция <code>number</code>
Входные данные	<ul style="list-style-type: none"> <li>• массив знаков, возвращённый функцией <code>signs</code></li> </ul>
Косвенные входные данные	—
Ожидаемый результат	Создание списка целых чисел длиной 5 в соответствии с условиями задачи.

Номер	Б10
Цель	Проверка на ошибку при неинициализированных входных данных
Тип	Негативный
Объект тестирования	Функция <code>number</code>
Входные данные	<ul style="list-style-type: none"> <li>• массив знаков, возвращённый функцией <code>signs</code>, с удалённым последним символом</li> </ul>
Косвенные входные данные	—
Ожидаемый результат	Ошибка при выполнении функции

Номер	Б11
Цель	Проверка работы при <code>diff = 2</code>
Тип	Позитивный
Объект тестирования	Функция <code>hideNumber</code>
Входные данные	<ul style="list-style-type: none"> <li>• <code>layoutNumbers</code> - список из 4 элементов (в данном случае, чтобы покрыть все элементы массива входных данных)</li> <li>• <code>diff = 2</code> - количество элементов, которые нужно скрыть</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Скрытие 2 случайных элементов из списка <code>layoutNumbers</code>, установка их текста в пустую строку и включение их доступности</li> </ul>

Номер	Б12
Цель	Проверка работы при $\text{diff} = 1$
Тип	Позитивный
Объект тестирования	Функция <code>hideNumber</code>
Входные данные	<ul style="list-style-type: none"> <li>• <code>layoutNumbers</code> - список из 3 элементов</li> <li>• <math>\text{diff} = 1</math> - количество элементов, которые нужно скрыть</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Скрытие 1 случайного элемента из списка <code>layoutNumbers</code>, установка его текста в пустую строку и включение его доступности</li> </ul>

Номер	Б13
Цель	Проверка работы при неинициализированных данных
Тип	Негативный
Объект тестирования	Функция <code>hideNumber</code>
Входные данные	<ul style="list-style-type: none"> <li>• <code>layoutNumbers</code> - список из 0 элементов</li> <li>• <math>\text{diff} = 1</math> - количество элементов, которые нужно скрыть</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Уведомление об ошибке и закрытие активности</li> </ul>

Класс «Pecesso»

Номер	Б14
Цель	Проверка стабильности работы при инициализированных данных
Тип	Позитивный
Объект тестирования	Функция <code>newGame</code>
Входные данные	<ul style="list-style-type: none"> <li>• <code>myPreferences</code> - объект <code>SharedPreferences</code> с установленным значением сложности ("<code>diff</code>"), например, 2</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Установка значения <code>end</code> в <code>true</code></li> <li>• Вызов функции <code>startGame</code> с параметром <code>diff</code>, равным значению сложности из <code>myPreferences</code> (в данном случае, 2)</li> </ul>

Номер	Б15
Цель	Проверка работы при неинициализированных данных
Тип	Негативный
Объект тестирования	Функция <code>newGame</code>
Входные данные	<ul style="list-style-type: none"> <li>• <code>myPreferences</code> - объект <code>SharedPreferences</code> с установленным значением сложности ("<code>diff</code>"), например, 0</li> </ul>
Ожидаемый результат	Уведомление об ошибке и закрытие активности



Номер	Б16
Цель	Проверка работы при инициализированных данных
Тип	Позитивный
Объект тестирования	Функция <code>startGame</code>
Входные данные	<ul style="list-style-type: none"> <li>• <code>diff = 2</code> - значение сложности</li> <li>• Существующий массив <code>cards</code> с элементами, представляющими идентификаторы кнопок</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Установка фона каждой кнопки из массива <code>cards</code> на изображение с рубашкой карты (<code>R.drawable.card_face</code>)</li> <li>• Установка таймера (<code>timer</code>) в значение, рассчитанное по формуле <math>90/diff</math></li> <li>• Установка значения <code>end</code> в <code>false</code></li> <li>• Установка значения <code>pair</code> в <code>0</code></li> </ul>

Номер	Б17
Цель	Проверка работы при неинициализированных данных
Тип	Негативный
Объект тестирования	Функция <code>startGame</code>
Входные данные	<ul style="list-style-type: none"> <li>• <code>diff = -1</code> - значение сложности</li> <li>• Существующий массив <code>cards</code> с элементами, представляющими идентификаторы кнопок</li> </ul>
Ожидаемый результат	Уведомление об ошибке и закрытие активности

Номер	Б18
Цель	Проверка работы при инициализированных данных
Тип	Позитивный
Объект тестирования	Функция <code>potasovka</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий массив <code>texturs</code> с 8 элементами</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Массив значений в случайном порядке из 8 элементов</li> </ul>

Номер	Б19
Цель	Проверка уникальности результата
Тип	Позитивный
Объект тестирования	Функция <code>potasovka</code>
Входные данные	<ul style="list-style-type: none"> <li>• массив <code>texturs</code> с 8 элементами</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Два массива, сгенерированных функцией отличаются друг от друга</li> </ul>

Номер	Б20
Цель	Проверка при неидеализированных входных данных
Тип	Негативный
Объект тестирования	Функция <code>potasovka</code>
Входные данные	<ul style="list-style-type: none"> <li>• массив <code>texturs</code> не инициализирован</li> </ul>
Ожидаемый результат	Уведомление об ошибке и закрытие активности

Номер	Б21
Цель	Проверка работы при инициализированных данных
Тип	Позитивный
Объект тестирования	Функция <code>startTimer</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий объект <code>handler</code></li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>timer</code></li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Установка значения текстового виджета <code>timer</code> в "0"</li> <li>• Запуск таймера, увеличивающего значение виджета каждую секунду, до момента, пока <code>end</code> не станет <code>true</code></li> </ul>

### 3.1.3 Подсистема прохождения мини-игры

Класс «`LineOfNumber`»

Номер	Б22
Цель	Проверка отчёта времени
Тип	Позитивный
Объект тестирования	Функция <code>startTimer</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий объект <code>handler</code></li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>timer</code> и текстом "5"</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Значение таймера при изменении меньше чем на секунду, не меняют текста объекта с идентификатором <code>timer</code>, при изменении на секунду, значение увеличивается на 1</li> </ul>

Номер	Б23
Цель	Проверка ошибки при неинициализированных входных данных
Тип	Негативный
Объект тестирования	Функция <code>startTimer</code>
Входные данные	<ul style="list-style-type: none"> <li>• Несуществующий объект <code>handler</code></li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>timer</code> и текстом "5"</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Уведомление об ошибки и закрытие активности</li> </ul>

Номер	Б24
Цель	Проверка работы при инициализированных данных
Тип	Позитивный
Объект тестирования	Функция <code>lineOfNumberCheck</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий массив <code>layoutNumbers</code> с 4 элементами</li> <li>• Существующий массив <code>layoutSings</code> с 3 элементами</li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>answer</code></li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Проверка, что все цифры в <code>layoutNumbers</code> были введены</li> <li>• Вычисление ответа по формуле, заданной значениями в <code>layoutNumbers</code> и <code>layoutSings</code></li> <li>• Сравнение вычисленного ответа с текстом в <code>layoutAnswer</code></li> <li>• Вывод сообщения "Всё правильно!" в случае совпадения и завершение игры вызовом функции <code>endGame</code></li> </ul>

Номер	Б25
Цель	Проверка обработки случая не введенных цифр
Тип	Негативный
Объект тестирования	Функция <code>lineOfNumberCheck</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий массив <code>layoutNumbers</code> с 4 элементами, из которых один элемент с пустым значением</li> <li>• Существующий массив <code>layoutSings</code> с 3 элементами</li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>answer</code></li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Вывод сообщения "Не все цифры были введены" через всплывающее окно <code>Toast</code></li> </ul>

Номер	Б25
Цель	Проверка обработки случая ответа, не совпадающего с правилами игры
Тип	Негативный
Объект тестирования	Функция <code>lineOfNumberCheck</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий массив <code>layoutNumbers</code> с 4 элементами, из которых один элемент с пустым значением</li> <li>• Существующий массив <code>layoutSings</code> с 3 элементами</li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>answer</code></li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Вывод сообщения "Ответ неправильный" через всплывающее окно <code>Toast</code></li> </ul>

Номер	Б26
Цель	Проверка при неинициализированных данных
Тип	Негативный
Объект тестирования	Функция <code>lineOfNumberCheck</code>
Входные данные	<ul style="list-style-type: none"> <li>• Несуществующий массив <code>layoutNumbers</code> с 4 элементами, из которых один элемент с пустым значением</li> <li>• Существующий массив <code>layoutSings</code> с 3 элементами</li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>answer</code></li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Уведомление об ошибке и закрытие активности</li> </ul>



Номер	Б27
Цель	Проверка завершения игры при инициализированных данных
Тип	Позитивный
Объект тестирования	Функция <code>endGame</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий флаг <code>end</code> с текущим значением <code>false</code></li> <li>• Существующий массив <code>layoutNumbers</code> с 4 элементами</li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>timer</code></li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Установка значения флага <code>end</code> в <code>true</code></li> <li>• Установка значения текстового виджета <code>timer</code> в <code>"0"</code></li> <li>• Отключение всех элементов массива <code>layoutNumbers</code></li> <li>• Вызов функции <code>createGame</code> с аргументом <code>diff</code>, полученным из <code>SharedPreferences</code></li> </ul>

Номер	Б28
Цель	Проверка при инициализированных данных
Тип	Негативный
Объект тестирования	Функция <code>endGame</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий флаг <code>end</code> с текущим значением <code>false</code></li> <li>• Существующий массив <code>layoutNumbers</code> с 4 элементами</li> <li>• Несуществующий текстовый виджет (<code>TextView</code>) с идентификатором <code>timer</code></li> </ul>
Ожидаемый результат	Уведомление об ошибке и прекращение активности

### Класс `Pecesso`

Номер	Б29
Цель	Проверка открытия карты
Тип	Позитивный
Объект тестирования	Функция <code>onCardClick</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий флаг <code>open</code> со значением <code>true</code></li> <li>• Коллекция <code>openCards</code> с меньше чем 2 элементами</li> <li>• Существующий массив <code>cards</code> с идентификаторами карт</li> <li>• Существующий объект <code>cardFace</code> - фон карты</li> <li>• Существующий объект <code>AnimationUtils</code></li> <li>• Существующий объект <code>Animation</code> с идентификатором <code>start_open</code></li> <li>• Существующий объект <code>Animation</code> с идентификатором <code>end_open</code></li> <li>• Существующий объект <code>Drawable</code> с фоном карты</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Добавление текущей карты в коллекцию <code>openCards</code></li> <li>• Запуск анимации открытия (<code>start_open</code>)</li> <li>• Проверка, что после анимации текущая карта имеет новый фон (<code>texturs[index]</code>) при совпадении с <code>cardFace</code>, иначе фон сменяется на <code>card_face</code></li> <li>• Запуск анимации закрытия (<code>end_open</code>) после завершения анимации открытия</li> </ul>

Номер	Б30
Цель	Проверка открытия второй несовпадающей карты
Тип	Позитивный
Объект тестирования	Функция <code>onCardClick</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий флаг <code>open</code> со значением <code>true</code></li> <li>• Коллекция <code>openCards</code> с меньше чем 2 элементами</li> <li>• Существующий массив <code>cards</code> с идентификаторами карт</li> <li>• Существующий объект <code>cardFace</code> - фон карты</li> <li>• Существующий объект <code>AnimationUtils</code></li> <li>• Существующий объект <code>Animation</code> с идентификатором <code>start_open</code></li> <li>• Существующий объект <code>Animation</code> с идентификатором <code>end_open</code></li> <li>• Существующий объект <code>Drawable</code> с фоном карты</li> <li>• Существующий объект <code>openCards</code> с размером 1</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Добавление текущей карты в коллекцию <code>openCards</code></li> <li>• Запуск анимации открытия (<code>start_open</code>)</li> <li>• Проверка, на отрицательное совпадение фона с открытой картой</li> <li>• Запуск анимации закрытия карт из <code>openCards</code> после завершения анимации открытия</li> </ul>

Номер	Б31
Цель	Проверка завершения игры с победой
Тип	Позитивный
Объект тестирования	Функция <code>endGame</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий флаг <code>end</code> со значением <code>true</code></li> <li>• Существующая переменная <code>pair</code> с значением 4</li> <li>• Существующий объект <code>Toast</code></li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Вывод сообщения о победе ("Вы выиграли!")</li> </ul>

Номер	Б32
Цель	Проверка завершения игры с поражением
Тип	Позитивный
Объект тестирования	Функция <code>endGame</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий флаг <code>end</code> со значением <code>true</code></li> <li>• Существующая переменная <code>pair</code> с значением отличным от 4</li> <li>• Существующий объект <code>Toast</code></li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Вывод сообщения о поражении ("Вы проиграли!")</li> </ul>

Номер	Б33
Цель	Проверка идентичности двух Drawable, используя константные состояния
Тип	Позитивный
Объект тестирования	Функция <code>areDrawablesIdentical</code>
Входные данные	<ul style="list-style-type: none"> <li>• Два существующих Drawable с одинаковыми константными состояниями</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Возвращение <code>true</code></li> </ul>

Номер	Б34
Цель	Проверка идентичности двух Drawable, используя изображения Bitmap
Тип	Позитивный
Объект тестирования	Функция <code>areDrawablesIdentical</code>
Входные данные	<ul style="list-style-type: none"> <li>• Два существующих Drawable с разными константными состояниями, но идентичными изображениями Bitmap</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Возвращение <code>true</code></li> </ul>

Номер	Б35
Цель	Проверка неидентичности двух Drawable
Тип	Негативный
Объект тестирования	Функция <code>areDrawablesIdentical</code>
Входные данные	<ul style="list-style-type: none"> <li>• Два существующих Drawable с разными константными состояниями и разными изображениями Bitmap</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Возвращение <code>false</code></li> </ul>

Номер	Б36
Цель	Проверка работы при инициализированных входных данных
Тип	Позитивный
Объект тестирования	Функция <code>startTimer</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий объект <code>handler</code></li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>timer</code></li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Установка значения текстового виджета <code>timer</code> в "0"</li> <li>• Запуск таймера, увеличивающего значение виджета каждую секунду, до момента, пока <code>end</code> не станет <code>true</code></li> </ul>

Номер	Б37
Цель	Проверка отчёта времени
Тип	Позитивный
Объект тестирования	Функция <code>startTimer</code>
Входные данные	<ul style="list-style-type: none"> <li>• Существующий объект <code>handler</code></li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>timer</code> и текстом "5"</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Значение таймера при изменении меньше чем на секунду, не меняют текста объекта с идентификатором <code>timer</code>, при изменении на секунду, значение увеличивается на 1</li> </ul>

Номер	Б38
Цель	Проверка ошибки при неинициализированных входных данных
Тип	Негативный
Объект тестирования	Функция <code>startTimer</code>
Входные данные	<ul style="list-style-type: none"> <li>• Несуществующий объект <code>handler</code></li> <li>• Существующий текстовый виджет (<code>TextView</code>) с идентификатором <code>timer</code> и текстом "5"</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Уведомление об ошибки и закрытие активности</li> </ul>



Номер	И1
Цель	Проверка передачи уровня сложности от «подсистемы выбора мини-игры» в «подсистему генерации мини-игры» при выборе игры «Числовой ряд»
Тип	Позитивный
Объект тестирования	Взаимодействие класса «TrainingActivity» и «LineOfNumber»
Входные данные	<ul style="list-style-type: none"> <li>• Нахождение в системе выбора мини-игры</li> <li>• Выбор значения «Числовой ряд» в поле spinnerGame</li> <li>• Выбор значения 1 в поле diff</li> <li>• Нажатие на кнопку «Играть»</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Создание экземпляра класса «LineOfNumber» со значением <math>diff = 1</math></li> </ul>

## 3.2 Интеграционное тестирование

В рамках этого приложения взаимодействие между подсистемами происходит в рамках передачи уровня сложности между «подсистемой выбора мини-игры» и «подсистемой генерации мини-игры». Эта зависимость будет тестироваться в рамках интеграционного тестирования.

Номер	И2
Цель	Проверка передачи уровня сложности от «подсистемы выбора мини-игры» в «подсистему генерации мини-игры» при выборе игры «Числовой ряд»
Тип	Позитивный
Объект тестирования	Взаимодействие класса «TrainingActivity» и «LineOfNumber»
Входные данные	<ul style="list-style-type: none"> <li>• Нахождение в системе выбора мини-игры</li> <li>• Выбор значения «Числовой ряд» в поле spinnerGame</li> <li>• Выбор значения 2 в поле diff</li> <li>• Нажатие на кнопку «Играть»</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Создание экземпляра класса «LineOfNumber» со значением <math>diff = 2</math></li> </ul>

Номер	ИЗ
Цель	Проверка передачи уровня сложности от «подсистемы выбора мини-игры» в «подсистему генерации мини-игры» при выборе игры «Перевертыши»
Тип	Позитивный
Объект тестирования	Взаимодействие класса «TrainingActivity» и «Pecsseso»
Входные данные	<ul style="list-style-type: none"> <li>• Нахождение в системе выбора мини-игры</li> <li>• Выбор значения «Перевертыши» в поле spinnerGame</li> <li>• Выбор значения 1 в поле diff</li> <li>• Нажатие на кнопку «Играть»</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Создание экземпляра класса «Pecsseso» со значением diff = 1</li> </ul>

Номер	И4
Цель	Проверка передачи уровня сложности от «подсистемы выбора мини-игры» в «подсистему генерации мини-игры» при выборе игры «Перевертыши»
Тип	Позитивный
Объект тестирования	Взаимодействие класса «TrainingActivity» и «Peccesso»
Входные данные	<ul style="list-style-type: none"> <li>• Нахождение в системе выбора мини-игры</li> <li>• Выбор значения «Перевертыши» в поле spinnerGame</li> <li>• Выбор значения 2 в поле diff</li> <li>• Нажатие на кнопку «Играть»</li> </ul>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Создание экземпляра класса «Peccesso» со значением diff = 2</li> </ul>

### 3.3 Аттестационное тестирование

Номер	A1
Цель	Проверка возможности выбора мини-игры «Числовой ряд»
Функциональное требование	«Выбор мини-игры» (п. 1), «Выбор сложности» (п. 2)
Тип	Позитивный
Сценарий действий	<ol style="list-style-type: none"> <li>1. Нажать на кнопку «Тренировка»</li> <li>2. Нажать на всплывающий список «Игра»</li> <li>3. Выбрать значение «Числовой ряд»</li> <li>4. Нажать на всплывающий список «Сложность»</li> <li>5. Выбрать значение «1»</li> <li>6. Нажать на кнопку «Играть»</li> </ol>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Создание экрана с игрой, реализованной классом "LineOfNumber" со сложностью 1</li> </ul>

Номер	A2
Цель	Проверка возможности выбора мини-игры «Перевертыши»
Функциональное требование	«Выбор мини-игры» (п. 1), «Выбор сложности» (п. 2)
Тип	Позитивный
Сценарий действий	<ol style="list-style-type: none"> <li>1. Нажать на кнопку «Тренировка»</li> <li>2. Нажать на всплывающий список «Игра»</li> <li>3. Выбрать значение «Перевертыши»</li> <li>4. Нажать на всплывающий список «Сложность»</li> <li>5. Выбрать значение «2»</li> <li>6. Нажать на кнопку «Играть»</li> </ol>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• Создание экрана с игрой, реализованной классом "LineOfNumber" со сложностью 2</li> </ul>

Номер	A3
Цель	Проверка возможности просмотра описания мини-игры «Перевёртыши»
Функциональное требование	«Выбор мини-игры» (п. 1), «Посмотреть рекорд в игре на каждом уровне сложности» (п. 3)
Тип	Позитивный
Сценарий действий	<ol style="list-style-type: none"> <li>1. Нажать на кнопку «Тренировка»</li> <li>2. Нажать на всплывающий список «Игра»</li> <li>3. Выбрать значение «Перевёртыши»</li> </ol>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• На экране будет предоставлена информация об рекорде и игре «Перевёртыши»</li> </ul>

Номер	A4
Цель	Проверка возможности просмотра описания мини-игры «Числовая ряд»
Функциональное требование	«Выбор мини-игры» (п. 1), «Посмотреть рекорд в игре на каждом уровне сложности» (п. 3)
Тип	Позитивный
Сценарий действий	<ol style="list-style-type: none"> <li>1. Нажать на кнопку «Тренировка»</li> <li>2. Нажать на всплывающий список «Игра»</li> <li>3. Выбрать значение «Числовой ряд»</li> </ol>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• На экране будет предоставлена информация об рекорде и игре «Числовой ряд»</li> </ul>



Номер	A5
Цель	Проверка возможности победного прохождения мини-игры «Числовой ряд»
Функциональное требование	<Пройти мини-игру» (п. 4)
Тип	Позитивный
Сценарий действий	<ol style="list-style-type: none"> <li>1. Нажать на кнопку «Тренировка»</li> <li>2. Нажать на всплывающий список «Игра»</li> <li>3. Выбрать значение «Числовой ряд»</li> <li>4. Нажать на всплывающий список «Сложность»</li> <li>5. Выбрать значение «2»</li> <li>6. Нажать на кнопку «Играть»</li> <li>7. Вставить цифры в пропущенные поля согласно правилам игры за 45 секунд</li> </ol>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• На экране появляется окно с текстом «Всё правильно!».</li> </ul>

Номер	А6
Цель	Проверка возможности проигрышного прохождения мини-игры «Числовой ряд»
Функциональное требование	<Пройти мини-игру» (п. 4)
Тип	Негативный
Сценарий действий	<ol style="list-style-type: none"> <li>1. Нажать на кнопку «Тренировка»</li> <li>2. Нажать на всплывающий список «Игра»</li> <li>3. Выбрать значение «Числовой ряд»</li> <li>4. Нажать на всплывающий список «Сложность»</li> <li>5. Выбрать значение «2»</li> <li>6. Нажать на кнопку «Играть»</li> <li>7. Ничего не делать в течение 45 секунд</li> </ol>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• На экране появляется окно с текстом «Вы проиграли!».</li> </ul>

Номер	A7
Цель	Проверка возможности победного прохождения мини-игры «Перевертыши»
Функциональное требование	<Пройти мини-игру> (п. 4)
Тип	Позитивный
Сценарий действий	<ol style="list-style-type: none"> <li>1. Нажать на кнопку «Тренировка»</li> <li>2. Нажать на всплывающий список «Игра»</li> <li>3. Выбрать значение «Перевертыши»</li> <li>4. Нажать на всплывающий список «Сложность»</li> <li>5. Выбрать значение «2»</li> <li>6. Нажать на кнопку «Играть»</li> <li>7. Вставить цифры в пропущенные поля согласно правилам игры за 45 секунд</li> </ol>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• На экране появляется окно с текстом «Всё правильно!».</li> </ul>

Номер	A8
Цель	Проверка возможности проигрышного прохождения мини-игры «Перевёртыши»
Функциональное требование	<Пройти мини-игру» (п. 4)
Тип	Негативный
Сценарий действий	<ol style="list-style-type: none"> <li>1. Нажать на кнопку «Тренировка»</li> <li>2. Нажать на всплывающий список «Игра»</li> <li>3. Выбрать значение «Перевёртыши»</li> <li>4. Нажать на всплывающий список «Сложность»</li> <li>5. Выбрать значение «2»</li> <li>6. Нажать на кнопку «Играть»</li> <li>7. Ничего не делать в течение 45 секунд</li> </ol>
Ожидаемый результат	<ul style="list-style-type: none"> <li>• На экране появляется окно с текстом «Вы проиграли!».</li> </ul>

Номер	H1
Цель	Проверка скорости генерации мини-игры «Перевёртыши»
Объект тестирования	экран «Pecsseso» (п. 4)
Входные данные Ожидаемый результат	<p>diff = 1</p> <ul style="list-style-type: none"> <li>• Генерация экрана «Pecsseso» произойдёт менее чем за 1 секунду.</li> </ul>

Номер	H2
Цель	Проверка скорости генерации мини-игры «Числовой ряд»
Объект тестирования	экран «LineOfNumber» (п. 4)
Входные данные Ожидаемый результат	<p>diff = 1</p> <ul style="list-style-type: none"> <li>• Генерация экрана «LineOfNumber» произойдёт менее чем за 1 секунду.</li> </ul>

### 3.4 Нагрузочное тестирование

Номер	Н3
Цель	Проверка реакции на множественное нажатие на объекты в игре «Перевертыши»
Объект тестирования	экран «Pecsesso», объекты типа Drawable (п. 4)
Входные данные Ожидаемый результат	<p>экран «Pecsseso»</p> <ul style="list-style-type: none"> <li>• При одновременном нажатии на множество объектов типа Drawable в игре не происходит ошибок и коллизий.</li> </ul>

Номер	Н4
Цель	Проверка реакции на множественное нажатие на объекты в игре «Числовой ряд»
Объект тестирования	экран «LineOfNumber», объекты типа TextView (п. 4)
Входные данные Ожидаемый результат	<p>экран «LineOfNumber»</p> <ul style="list-style-type: none"> <li>• При одновременном нажатии на множество объектов типа TextView в игре не происходит ошибок и коллизий.</li> </ul>

# Глава 4

## Примеры тестов

### 4.1 Блочный тест Б23

```
fun testStartTimer() = runBlocking {  
    // Подготовка тестовых данных  
    logicLON.end = false  
    logicLON.timer = 0  
  
    // Установка поведения мока  
    Mockito.`when`(mockHandler.postDelayed(Mockito.any(),Mockito.anyLong())).then{  
        (it.arguments[0] as Runnable).run()  
        null  
    }  
  
    // Запуск тестируемого метода  
    logicLON.startTimer()  
  
    // Ждем, чтобы таймер мог увеличиться  
    delay(1100)  
  
    // Проверки  
    assertTrue(logicLON.timer >= 1)  
  
    // Завершение работы таймера
```

```

logicLON.end = true

// Ждем, чтобы убедиться, что таймер больше не увеличивается
val initialTimerValue = logicLON.timer
delay(1100)
assertEquals(initialTimerValue, logicLON.timer)
}

```

## 4.2 Интеграционный тест И1

```

@RunWith(AndroidJUnit4::class)
class TrainingActivityTest {

    @Rule
    @JvmField
    val activityTestRule = ActivityTestRule(TrainingActivity::class.java)

    @Test
    fun testDifficultyTransferToLineOfNumber() {
        // Выполняем действия, симулирующие взаимодействие пользователя с приложением
        onView(withId(R.id.spinnerGame)).perform(click())
        onData(allOf('is'(instanceOf(String::class.java)), 'is'("Числовой ряд")))
            .perform(click())

        onView(withId(R.id.diff)).perform(click())
        onData(allOf('is'(instanceOf(String::class.java)), 'is'("1")))
            .perform(click())

        onView(withId(R.id.playButton)).perform(click())

        // Проверяем, был ли создан экземпляр LineOfNumber с правильным уровнем сложности
        intended(allOf(
            hasComponent(LineOfNumber::class.java.name),
            hasExtra(LineOfNumber.EXTRA_DIFFICULTY, 1)
        ))
    }
}

```



))  
}  
}

## Глава 5

# Поккрытие кода тестами

Встроенный в Android Studio инструмент coverage оценивает поккрытие кода тестами в 78%. Тестами не были поккрыты генераторы экземпляров классов и главное меню.

## Глава 6

### Журнал тестирования

Номер	Дата	Результат	Номер ошибки в журнале
Блочное тестирование			
Б1	15.12.2023	Пройден	—
Б2	15.12.2023	Пройден	—
Б2	15.12.2023	Пройден	—
Б4	15.12.2023	Пройден	—
Б5	15.12.2023	Пройден	—
Б6	15.12.2023	Пройден	—
Б7	15.12.2023	Пройден	—
Б8	15.12.2023	Пройден	—
Б9	15.12.2023	Пройден	—
Б10	15.12.2023	Пройден	—
Б11	15.12.2023	Пройден	—
Б12	15.12.2023	Пройден	—
Б13	15.12.2023	Пройден	—
Б14	15.12.2023	Пройден	—
Б15	15.12.2023	Пройден	—
Б16	15.12.2023	Пройден	—
Б17	15.12.2023	Пройден	—
Б18	15.12.2023	Пройден	—
Б19	15.12.2023	Пройден	—
Б20	15.12.2023	Пройден	—
Б21	15.12.2023	Пройден	Отчёт об ошибке №1
Б22	15.12.2023	Пройден	—
Б23	15.12.2023	Пройден	—
Б24	15.12.2023	Пройден	—
Б25	15.12.2023	Пройден	—
Б26	15.12.2023	Пройден	—
Б27	15.12.2023	Пройден	—
Б28	15.12.2023	Пройден	—
Б29	15.12.2023	Пройден	—
Б30	15.12.2023	Пройден	—
Б31	15.12.2023	Пройден	—

Номер	Дата	Результат	Номер ошибки в журнале
Блочное тестирование			
Б32	15.12.2023	Пройден	—
Б33	15.12.2023	Пройден	—
Б34	15.12.2023	Пройден	—
Б35	15.12.2023	Пройден	—
Б36	15.12.2023	Пройден	—
Б37	15.12.2023	Пройден	—
Б38	15.12.2023	Пройден	—
Интеграционное тестирование			
И1	15.12.2023	Пройден	—
И2	15.12.2023	Пройден	—
И3	15.12.2023	Пройден	—
И4	15.12.2023	Пройден	—
Аттестационное тестирование			
А1	15.12.2023	Пройден	—
А2	15.12.2023	Пройден	—
А3	15.12.2023	Пройден	—
А4	15.12.2023	Пройден	Отчёт об ошибке №2
А5	15.12.2023	Пройден	—
А6	15.12.2023	Пройден	—
А7	15.12.2023	Пройден	—
А8	15.12.2023	Пройден	—

## Глава 7

### Журнал ошибок

№ отчёта об ошибке	1
Дата составления отчёта	15.12.2023
Номер теста	Б21
Ожидаемый результат	Установка значения текстового виджета timer в "0"
Фактический результат	Установка значения текстового виджета timer в "90"
Воспроизводимость	Всегда
Приоритет	Высокий
Статус	Решено

№ отчёта об ошибке	2
Дата составления отчёта	15.12.2023
Номер теста	A4
Ожидаемый результат	На экране будет предоставлена информация об рекорде и игре «Числовой ряд»
Фактический результат	На экран не выводится информация о рекорде и игре «Числовой ряд»
Воспроизводимость	Всегда
Приоритет	Низкий
Статус	Решено

## Глава 8

# Результаты тестирования

В рамках дисциплины «Верификация программного обеспечения» были протестированы функциональные возможности программной системы «Развивариум», а именно: приложения для тренировки и развития умственных способностей на языке программирования Kotlin, в среде разработки Android Studio. На этапах модульного и интеграционного тестирования, проводимого посредством автоматизированного тестирования, не было выявлено критических ошибок в функционировании рассмотренных методов системы. На этапе аттестационного тестирования, проводимого посредством ручного тестирования, критические ошибки также выявлены не были. На этапе тестирования безопасности, проводимого посредством автоматизированного тестирования, не было выявлено ошибок, связанных с возможностью нарушения информационной целостности приложения. Таким образом, информационная система «Развивариум» успешно прошла тестирование.