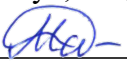


ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук  
Образовательная программа бакалавриата «Программная инженерия»

СОГЛАСОВАНО  
Руководитель проекта,  
Доцент департамента  
программной инженерии  
факультета компьютерных  
наук, канд. техн. наук  
 / С.Л. Макаров /  
«11» апреля 2022 г.

УТВЕРЖДАЮ  
Академический руководитель  
образовательной программы  
«Программная инженерия»  
проф, канд. техн. наук  
\_\_\_\_\_/В.В. Шилов  
«\_\_» \_\_\_\_\_ 2022 г.

Андроид-приложение "Игра Переливания"  
Пояснительная записка  
ЛИСТ УТВЕРЖДЕНИЯ  
RU.17701729.05.03-01 81 01-1-ЛУ

Исполнитель:

студент группы БПИ197



\_\_\_\_\_/А.Д.Сальникова /

«07» \_\_\_\_\_ апреля 2022 г.

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

Москва 2022

УТВЕРЖДЕН  
RU.17701729.05.03-01 81 01-1-ЛУ

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

Андроид-приложение "Игра Переливания"

Пояснительная записка

RU.17701729.05.03-01 81 01-1

Листов 21

Москва 2022

## **АННОТАЦИЯ**

В данном программном документе приведена пояснительная записка к программе «app-release.apk» («Андроид-приложение "Игра Переливания" (Transfusions Game Android Application)»), предназначенной для игры на мобильном устройстве в цифровой аналог математической задачи на переливания.

В разделе «Введение» указано наименование программы и документ, на основании которого ведется разработка.

В разделе «Назначение и область применения» указано функциональное и эксплуатационное назначение программы, а также краткая характеристика области применения программы.

В разделе «Технические характеристики» есть следующие подразделы: описание математической задачи на переливания, на основании которой разработана программа; математическая составляющая нахождения решения такой задачи; алгоритмы функционирования программы и ее интерфейс; выбор технических и программных средств.

В разделе «Ожидаемые технико-экономические показатели» указана предполагаемая потребность и полезность разработки, преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами не указаны, так как аналоги найдены не были.

Настоящий документ разработан в соответствии с требованиями:

- 1) ГОСТ 19.101-77 Виды программ и программных документов [1];
  - 2) ГОСТ 19.102-77 Стадии разработки [2];
  - 3) ГОСТ 19.103-77 Обозначения программ и программных документов [3];
  - 4) ГОСТ 19.104-78 Основные надписи [4];
  - 5) ГОСТ 19.105-78 Общие требования к программным документам [5];
  - 6) ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом [6];
  - 7) ГОСТ 19.404-79 Пояснительная записка. Требования к содержанию и оформлению [7].
- Изменения к данному Техническому заданию оформляются согласно ГОСТ 19.603-78 [8], ГОСТ 19.604-78 [9].

**Оглавление**

<b>1</b>	<b>Введение.....</b>	<b>4</b>
1.1.	Наименование программы .....	4
1.2.	Документ, на основании которого ведется разработка .....	4
<b>2</b>	<b>Назначение и область применения .....</b>	<b>5</b>
2.1.	Назначение программы .....	5
2.2.	Краткая характеристика области применения .....	5
<b>3</b>	<b>Технические характеристики .....</b>	<b>6</b>
3.1.	Описание математической задачи на переливания .....	6
3.2.	Математическое решение задачи .....	7
3.3.	Алгоритмы функционирования программы и ее интерфейс .....	9
3.4.	Выбор технических и программных средств .....	12
<b>4</b>	<b>Ожидаемые технико-экономические показатели .....</b>	<b>13</b>
	Источники, использованные при разработке .....	14
	Приложение 1 Таблица с описанием классов и их функциональным назначением.....	15
	Приложение 2 Описание и функциональное описание методов .....	16

## **1 ВВЕДЕНИЕ**

### **1.1. Наименование программы**

Название программы: Андроид-приложение "Игра Переливания" (Transfusions Game Android Application). Далее программа "игра Переливания", исполняемый файл app-release.apk.

### **1.2. Документ, на основании которого ведется разработка**

Программа разработана в рамках выполнения курсовой работы — Андроид-приложение "Игра Переливания", в соответствии с учебным планом 3 курса бакалавриата направления 09.03.04 «Программная инженерия».

Основанием для разработки является учебный план подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденная академическим руководителем программы тема курсового проекта.

Наименование темы разработки Андроид-приложение "Игра Переливания" (Transfusions Game Android Application).

## **2 НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ**

### **2.1. Назначение программы**

Программа представляет собой 2D игру, в которой реализовывается математическая задача на переливание жидкости по сосудам разного объема для получения заданного количества жидкости в одном из сосудов. Правила не ограничивают появление жидкости в сосудах из внешнего источника, но каждый сосуд ограничен своим объемом.

Предполагается, что будет реализовано две основных функции для приложения:

решение существующей в приложении задачи на переливания с проверкой корректности решения;

проверка задачи из вводимых пользователем значений объема сосудов и желаемого объема на наличие решения в соответствии с правилами переливаний.

Приложение предназначено для пользователей разных возрастов на мобильном устройстве. Доступ в Интернет для игры не требуется.

### **2.2. Краткая характеристика области применения**

Разрабатываемый продукт предназначен для досуга, развития логики и алгоритмического мышления. Предполагаемая потребность обуславливается тем фактом, что на данный момент нет приложения, реализующего рассматриваемую математическую задачу. Предлагаемые уровни позволяют пользователю освоить и проверить способы переливаний между сосудами. Также программу можно применить для проверки пользовательского уровня на наличие решения.

### 3 ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

#### 3.1. Описание математической задачи на переливания

Математическая задача на переливания - это классическая задача, наиболее часто встречающаяся у школьников младших или средних классов. Она относится к числу широко известных головоломок. Это связано так же с тем, что такой тип задач имеет многовековую историю, но все еще не теряет актуальности и в наше время. Одной из наиболее старинных известных задач на переливание является задача Пуассона. В начале 19 века он сформулировал такую задачу: "Человек имеет в бочонке 12 пинт вина и хочет отмерить половину, но у него нет сосуда в 6 пинт, но есть два пустых сосуда объемом 5 и 8 пинт. Сможет ли он разделить вино?". В чуть более современной формулировке это звучит: "Как из полного сосуда емкостью 12л отмерить 6л, используя два пустых сосуда объемом 5 и 8л?".

В настоящее время такие задачи встречаются практически в каждом сборнике математических головоломок, но среди всех задач можно выделить общие признаки, которые позволяют разделить такие задачи на группы.

Первая классификация более очевидная — это классификация по числу сосудов для жидкости. Среди просмотренных задач были выделены задачи с двумя сосудами и с тремя сосудами.

Вторая классификация определяется замкнутостью системы. В первом случае рассматривается открытая система. То есть помимо нескольких сосудов существует бесконечный источник, в котором можно неограниченно набирать воду, а также сливать в него воду. Во втором случае система замкнута. Это означает, что система ограничена только имеющимися сосудами конкретного объема и объемом жидкости, которое по условию есть в этих сосудах.

Сам процесс переливания между сосудами ограничен следующими правилами:

- сосуд опустошается до конца или до наполнения сосуда для переливания;
- сосуд наполняется до конца или до опустошения сосуда, из которого наливается жидкость.

Это значит, что, например, в открытой системе при наполнении сосуда из бесконечного источника конечный результат зависит только от объема наполняемого сосуда. После наполнения даже при наличии изначально в сосуде какого-то объема в итоге сосуд будет заполнен полностью. Так же и при опустошении сосуда в бесконечный источник. Сколько бы жидкости не было в сосуде изначально сосуд будет полностью опустошен.

При переливании между сосудами появляются остатки в сосудах за счет разниц в объемах, что и позволяет решать такие задачи. То есть при наполнении второго сосуда из первого есть два варианта:

- остаток жидкости в первом сосуде меньше или равен свободному объему второго сосуда. Тогда первый сосуд полностью опустошается, а второй сосуд наполняется ровно на тот объем, который был изначально в первом сосуде. Пример: переливание из полного сосуда 5л в пустой сосуд 8л приведет к распределению 0л в 5-литровом сосуде и 5л в 8-литровом сосуде;

- остаток жидкости в первом сосуде больше свободного объема второго сосуда. Тогда второй сосуд полностью наполняется, а первый сосуд опустошается ровно на объем равный свободному до переливания объему второго сосуда. Пример: переливание из полного сосуда 8л перелить воду в пустой сосуд 5л приведет к распределению 3л в 8-литровом сосуде, второй сосуд заполнен полностью на 5л.

Тогда для примера решения подобной задачи рассмотрим задачу Пуассона. Для наглядности решения воспользуемся таблицей, где сверху указаны сосуды, а ниже таблица остатка жидкости в каждом сосуде после каждого переливания:

12л	8л	5л
12	0	0
4	8	0
4	3	5
9	3	0
9	0	3
1	8	3
1	<b>6</b>	5
<b>6</b>	<b>6</b>	0

### 3.2. Математическое решение задачи

Любая математическая и тем более логическая задача имеет несколько вариантов решений, которые могут варьироваться по своей трудоемкости и эффективности. Рассмотрим некоторые из них, которые были рассмотрены в данной задаче.

Очевидно, что наиболее известный способ решения любой задачи — это решение подбором или также можно назвать его методом рассуждений[4]. Действительно, используя последовательные рассуждения можно прийти к какому-то решению данного типа задач, но следует отметить, что такой метод может не привести к решению, так как основан только на логике конкретного человека в конкретный момент времени.



Методу рассуждений будет соответствовать еще и метод таблиц[4]. Конечно, например в задаче Пуассона в качестве решения приведена таблица, но основным отличием от метода рассуждений является наглядность, а не эффективность. По сути, таблица является графическим представлением метода рассуждений.

Тогда для решения может подойти метод графа или метод полного перебора [2]. До сих пор такое решение для такого типа задач всегда достигает цели, но в то же время это одно из наиболее трудоемких и время затратных решений для любой логической задачи. Подобное решение будет представлять из себя дерево, каждое разветвление вершины такого дерева может содержать от одного до шести ветвлений. Понятно, что при таком ветвлении сложно эффективно решить задачу, так как неизвестно сколько слоев дерева потребуется для решения.

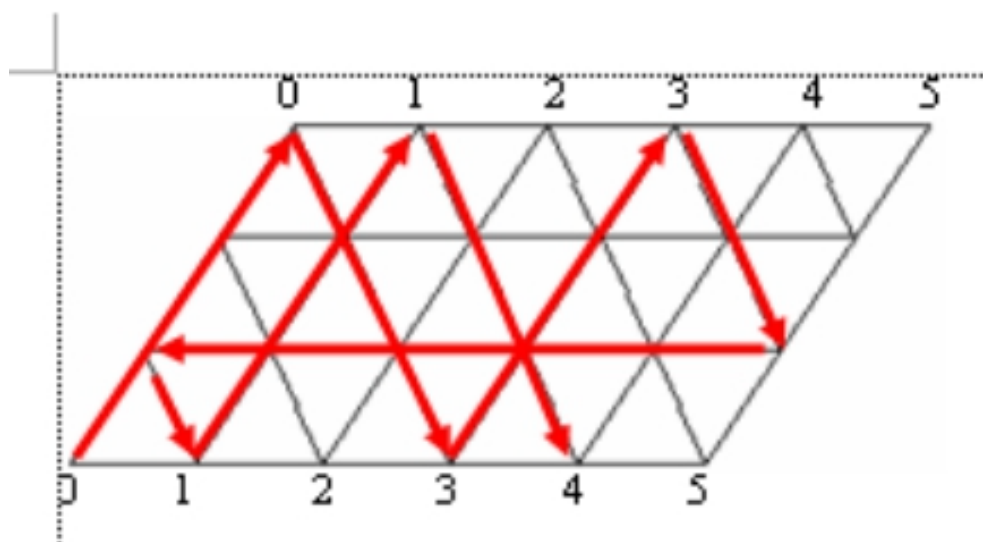
Итоговым решением для программы был выбран метод бильярда или параллелограмма [4]. Этот универсальный метод для решения некоторых типов логических задач был сформулирован еще в начале прошлого века Яковом Перельманом.

За основу этого метода для данной задачи берется параллелограмм с углом 60 градусов и сторонами равными объему двух меньших сосудов. Таким образом, по одной из сторон параллелограмма с так называемой точки (0,0) запускается луч, который соответственно будет отражаться от сторон параллелограмма. Так как угол падения в таком случае равен углу отражения, а параллелограмм берется с углом в 60 градусов, то в результате отражения луч будет поочередно отражаться от различных сторон под углом в 60 градусов. Луч проходит по какой-то траектории и возвращается в начальную точку, так как правый край луча поднимается с каждой итерацией по параллелограмму и в итоге пришел бы в верхнюю точку, что означает, что до этого он пришел в правую нижнюю точку, а значит следующим отражением возвращается в точку (0,0). Применение этого метода к данной задаче означает, что все точки, в которые луч попал достижимы в такой задаче, остальные объемы получить нельзя.

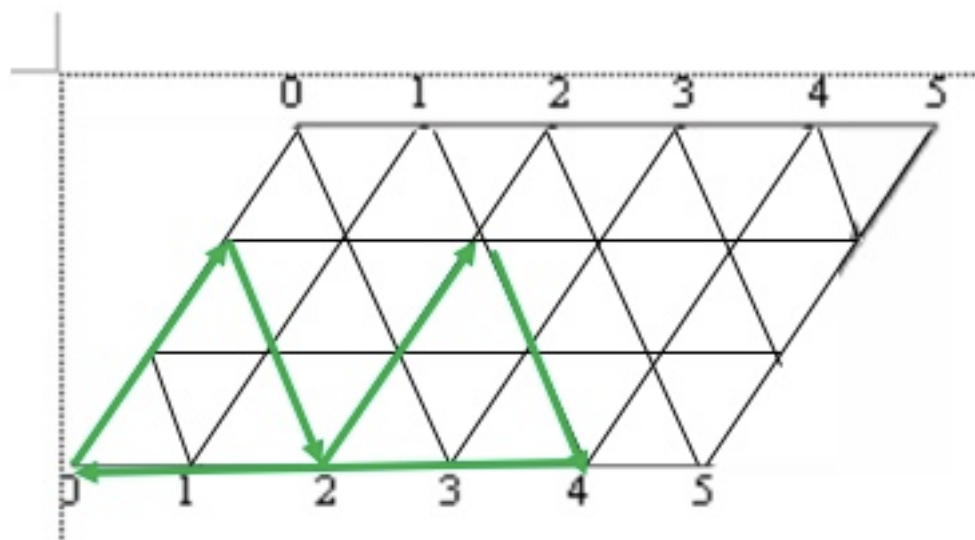
Докажем, что такой метод действительно можно применить к текущей задаче. Пусть мы пускаем луч по меньшей стороне параллелограмма. Это аналогично наполнению меньшего сосуда из большего в случае трех сосудов или из бесконечного источника в случае двух сосудов. Затем при отражении мы приходим к нулю по меньшей стороне и значению на большей стороне, равному длине меньшей стороны. Это аналогично переливанию из меньшего сосуда в оставшийся. Повторение этой итерации приводит к тому, что больший сосуд полностью наполняется, а в меньшем остается какой-то остаток, который отличен от объема этого сосуда. Следующее отражение получается параллельным длинной стороне, то есть аналогично сливанию большего сосуда. Затем начинается новый цикл. Таким образом,

отражение луча от сторон параллелограмма соответствует переливанию жидкости между двумя сосудами. Значит, если для метода параллелограмма не выполняется прохождение точки с координатами  $(x, y)$ , то невозможно получить объемы  $x$  и  $y$  в сосудах, соответствующих сторонам параллелограмма. Отличие замкнутой системы с тремя сосудами здесь в том, что для такой системы будет решение задачи даже если луч не проходит  $(x, y)$ , где  $x$  или  $y$  нужный объем, так как каждой точке  $(a, b)$  соответствует объем третьего сосуда  $max(a - b, 0)$ , который может быть равен  $x$  или  $y$ .

Для примера рассмотрим два сосуда 3л и 5л, в которые нужно набрать 4л. Пускаем луч(красный) по меньшей стороне и смотрим на его отражение. Видим, что луч придет в точку  $(4, 0)$ , значит можно такими сосудами отмерить 4л.



Для примера, если мы возьмем одинаковые объемы сосудов или кратные друг другу, то получить все объемы будет нельзя. Пусть это будет зеленый луч для параллелограмма 2 на 4, то есть соответствовать сосудам 2л и 4л. Из рисунка понятно, что с помощью таких сосудов невозможно отмерить объемы отличные от этих сосудов.



### 3.3. Алгоритмы функционирования программы и ее интерфейс

Программа включает в себя 9 различных экранов. Экраны, содержащие в себе информацию — это экраны правил и об авторах. Экраны с выбором — это главное меню, меню выбора режима игры, меню выбора уровня режима двух сосудов и меню выбора уровня режима трех сосудов. Основными являются экраны игрового поля для двух и для трех сосудов, а также экран проверки пользовательского уровня.

Между этими экранами осуществляется навигация по соответствующим кнопкам. И главного меню можно попасть на экран выбора режима игры, экран правил и экран об авторах. Из окон правил и об авторах можно попасть только обратно в главное меню. Из окна выбора режима игры можно попасть в выбор уровней соответствующего режима, а также обратно в главное меню. Из окон выбора уровней текущего режима можно попасть обратно в окно выбора режима, перейти на игровой экран соответствующего режима с выбранными условиями уровня или на окно проверки пользовательской задачи.

В окне игрового поля с двумя сосудами можно взаимодействовать с двумя ведрами заданного объема, которые изначально пустые, а также с кнопками опустошить и наполнить. Соответственно для этого окна реализуются следующие алгоритмы

- опустошение сосуда;
- наполнение сосуда;
- переливание между сосудами;

Переливание между сосудами сравнивает объем оставшейся жидкости первого сосуда со свободным объемом второго сосуда. Затем в соответствии с правилом игры переливает возможный объем жидкости. Для этого сравниваются свободный объем и остаток в первом сосуде. Если остаток больше, то второй сосуд заполняется полностью, а в первом остается разность между остатком и свободным объемом. Если остаток равен свободному объему, то второй сосуд полностью наполняется, а первый полностью опустошается. Если же свободный объем больше, то первый сосуд опустошается полностью, а второй наполняется на весь остаток.

Так рассмотрим пример переливания из заполненного 8л ведра в опустошенное 3л ведро. Свободный объем в 3л ведре равен 3л. Остаток в 8л ведре равен 8л. Значит остаток больше свободного объема. По условию в таком случае 3л ведро заполняется полностью, а в 8л остается разность между свободным объемом и остатком, то есть  $8-3 = 5$ л. При осуществлении этого действия изображения ведер также обновились на необходимые объемы.



В окне с тремя сосудами соответственно реализован только алгоритм переливания между сосудами.

В окне проверки реализованы:

- алгоритм проверки для открытой системы из двух сосудов, который реализует метод параллелограмма из математического решения задачи;

Для этого используем такой алгоритм:

- 1) Наполняем пустое меньшее ведро;
- 2) Переливаем из него воду по правилам переливаний в большее ведро;
- 3) Если большее ведро наполнилось, то опустошаем его;
- 4) Переливаем остаток из меньшего ведра в большее;
- 5) Возвращаемся в начало алгоритма.

Алгоритм заканчивается, когда объем в одном из ведер равен необходимому объему или если объем обоих сосудов стал равен нулю.

- алгоритм проверки для замкнутой системы из трех сосудов, который так же реализует метод параллелограмма, но ограничивает его из-за ситуации, когда суммарный объем двух меньших сосудов превосходит объем большего сосуда.

При таком условии повторяем алгоритм двух сосудов при условии, что на 1 шаге проверяем, что объем остатка самого большого ведра больше, чем меньшее ведро иначе переливаем в

меньшее ведро полностью остаток самого большого ведра.

Также алгоритм завершается при условии, что разность объема самого большого сосуда и суммарного объема двух других равна необходимому объему.

### **3.4. Выбор технических и программных средств**

Для удобного использования программы и корректного функционирования рекомендованные параметры устройства:

- не менее 300 МБ свободного пространства;
- система Android с версией не ниже Android8.0 Oreo;
- разрешение экрана около 2340x1080.

Для разработки были использованы JetBrainsIntelliJ Idea.2020.2.1 и AndroidStudio.4.1.3.

#### **4 ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ**

Разрабатываемый продукт предназначен для досуга, развития логики и алгоритмического мышления. Предполагаемая потребность обуславливается тем фактом, что на данный момент нет приложения, реализующего рассматриваемую математическую задачу. Поэтому программа будет интересна как людям, которые хотят попробовать научиться решать эту несложную математическую задачу, но при этом цифровая версия не требует от них ведения собственных записей решения, а наглядно представляет текущее состояние сосудов. Также приложение может быть интересно создателям таких задач, так как позволит получить условия задачи без необходимости решать ее. Не было найдено существующих аналогов. Разрабатываемое приложение является решением математической задачи, для которой нет цифровых версий.

**ИСТОЧНИКИ, ИСПОЛЬЗОВАННЫЕ ПРИ РАЗРАБОТКЕ**

- 1) Сборник математических задач по переливанию [Электронный ресурс], <http://mmmf.msu.ru/archive/20122013/z5/z5011212.html>. (Дата обращения 31.01.2022)
- 2) Сборник математических задач по переливанию [Электронный ресурс], [https://www.problems.ru/view\\_by\\_subject\\_new.php?parent=207&start=0](https://www.problems.ru/view_by_subject_new.php?parent=207&start=0). (Дата обращения 31.01.2022)
- 3) Документация к изучению среды Android Studio [Электронный ресурс], <https://developer.android.com/guide?hl=ru>. (Дата обращения 31.01.2022)
- 4) Иванов Б. Н. Дискретная математика Алгоритмы и программы: Учеб. пособие /Б. Н. Иванов. —М.: Лаборатория Базовых Знаний, 2003.
- 5) ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. / Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 6) ГОСТ 19.505-79 Руководство оператора. Требования к содержанию и оформлению. / Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 7) ГОСТ 19.404-79 Пояснительная записка. Требования к содержанию и оформлению. / Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 8) ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению. / Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 9) ГОСТ 19.401-78 Текст программы. Требования к содержанию и оформлению. / Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

**Таблица 1. Таблица с описанием классов и их функциональным назначением**

<b>Класс</b>	<b>Назначение</b>
MainActivity	Класс главного экрана, задает навигацию для главного меню
rules	Класс окна правил, содержит навигацию к главному меню
authors	Класс окна авторов, содержит навигацию к главному меню
choosing	Класс выбора режима игры
choose2	Класс выбора уровня для режима игры с двумя сосудами
choose3	Класс выбора уровня для режима игры с тремя сосудами
game2	Основной класс, описывающий алгоритм игры с двумя сосудами
game3	Основной класс, описывающий алгоритм игры с тремя сосудами
Check	Основной класс, описывающий алгоритм проверки введенного условия



**Описание и функциональное описание методов**

**Таблица 2. Описание методов класса MainActivity**

Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна главного меню
onClickRules	public	void	View v	Переход к окну правил
onClickAuthors	public	void	View v	Переход к окну авторов
onClickPlay	public	void	View v	Переход к окну выбора режима

**Таблица 3. Описание методов класса rules**

Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна правил
onClickBack	public	void	View v	Переход к главному меню

**Таблица 4. Описание методов класса authors**

Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна об авторах
onClickBack	public	void	View v	Переход к главному меню

**Таблица 5. Описание методов класса choosing**

Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна выбора режима игры

onClickBack	public	void	View v	Переход к главному меню
onClickTwo	public	void	View v	Переход к окну выбора уровня режима двух сосудов
onClickThree	public	void	View v	Переход к окну выбора уровня режима трех сосудов

**Таблица 6. Описание методов класса choose2**

Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна выбора уровня в режиме двух сосудов
onClickBack	public	void	View v	Переход к окну выбора режима игры
onClickPlay1	public	void	View v	Переход к игровому окну первого уровня режима двух сосудов
onClickPlay2	public	void	View v	Переход к игровому окну второго уровня режима двух сосудов
onClickPlay3	public	void	View v	Переход к игровому окну третьего уровня режима двух сосудов
onClickPlay4	public	void	View v	Переход к окну проверки условий режима двух сосудов

**Таблица 7. Описание методов класса choose3**

Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна выбора уровня в режиме трех сосудов
onClickBack	public	void	View v	Переход к окну выбора режима игры

onClickPlay1	public	void	View v	Переход к игровому окну первого уровня режима трех сосудов и передача условий уровня
onClickPlay2	public	void	View v	Переход к игровому окну второго уровня режима трех сосудов и передача условий уровня
onClickPlay3	public	void	View v	Переход к игровому окну третьего уровня режима трех сосудов и передача условий уровня
onClickPlay4	public	void	View v	Переход к окну проверки условий режима трех сосудов

**Таблица 8. Описание методов класса game2**

Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание игрового окна в режиме двух сосудов и задание базовых переменных по переданному условию уровня
onClickBack	public	void	View v	Переход к окну выбора уровня в режиме двух сосудов
onClickSmallBucket	public	void	View v	Реакция на нажатие кнопки малого ведра
onClickBigBucket	public	void	View v	Реакция на нажатие кнопки большого ведра
onClickFull	public	void	View v	Реакция на нажатие кнопки опустошить
onClickEmpty	public	void	View v	Реакция на нажатие кнопки наполнить
transfuse	public	void	-	Алгоритм переливания
end	public	void	-	Завершение уровня

**Таблица 9. Описание методов класса game3**

Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание игрового окна в режиме трех сосудов и задание базовых переменных по переданному условию уровня
onClickBack	public	void	View v	Переход к окну выбора уровня в режиме трех сосудов
onClickSmallBucket	public	void	View v	Реакция на нажатие кнопки малого ведра
onClickMediumBucket	public	void	View v	Реакция на нажатие кнопки среднего ведра
onClickBigBucket	public	void	View v	Реакция на нажатие кнопки большого ведра
transfuse	public	void	-	Алгоритм переливания
end	public	void	-	Завершение уровня

**Таблица 10. Описание методов класса Check**

Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна проверки в режиме выбранном при выборе проверки
onClickBack	public	void	View v	Переход к окну выбора режима игры
onClickCheck	public	void	View v	Реакция на нажатие кнопки проверить и запуск проверки введенных условий

RU.17701729.05.03-01 81 01-1  
**ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

[illegible]