Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Волгоградский государственный технический университет»

Факультет электроники и вычислительной техники

Кафедра «Программное обеспечение автоматизированных систем»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовой работе**

по дисциплине «Объектно-ориентированный анализ и программирование»

на тему: «Проектирование и реализация программы с использованием объектно-ориентированного подхода»

(индивидуальное задание – вариант № 21\_02)

Студент: Шеху А.У.

Группа: ПрИн-367

Работа зачтена с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ «      » \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20    г.

Руководитель проекта, нормоконтроллер \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Литовкин Д.В.

Волгоград 2022 г.

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Волгоградский государственный технический университет»

Факультет электроники и вычислительной техники

Направление 09.03.04 «Программная инженерия»   
Кафедра «Программное обеспечение автоматизированных систем»

Дисциплина «Объектно-ориентированный анализ и программирование»

Утверждаю

Зав. кафедрой \_\_\_\_\_\_\_\_\_\_ Орлова Ю.А.

**ЗАДАНИЕ**

**на курсовую работу**

Студент: Шеху А.У.

Группа: ПрИн-367

1. Тема: «Проектирование и реализация программы с использованием объектно-ориентированного подхода» (индивидуальное задание – вариант №21\_02)

Утверждена приказом от «24» января 2021г. № 101-ст

2. Срок представления работы к защите « 04 »   июня  2022 г.

3. Содержание пояснительной записки:

формулировка задания, требования к программе, структура программы, типовые процессы в программе, человеко-машинное взаимодействие, код программы и модульных тестов

4. Перечень графического материала:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5. Дата выдачи задания «12» февраля 2022 г.

Руководитель проекта: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Литовкин Д.В.

Задание принял к исполнению: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Шеху А.У.

«12» февраля 2022 г.

**Содержание**

[1. Формулировка задания 4](#_Toc99064212)

[2. Нефункциональные требования 4](#_Toc99064213)

[3. Первая итерация разработки 5](#_Toc99064214)

[3.1. Формулировка упрощённого варианта задания 5](#_Toc99064215)

[3.2. Функциональные требования (сценарии) 5](#_Toc99064216)

[3.3. Словарь предметной области 11](#_Toc99064217)

[3.4. Структура программы на уровне классов 11](#_Toc99064218)

[3.5. Типовые процессы в программе 11](#_Toc99064219)

[3.6. Человеко-машинное взаимодействие 11](#_Toc99064220)

[3.7. Реализация ключевых классов 11](#_Toc99064221)

[4. Вторая итерация разработки 11](#_Toc99064222)

[4.1. Функциональные требования (сценарии) 11](#_Toc99064223)

[4.2. Словарь предметной области 11](#_Toc99064224)

[4.3. Структура программы на уровне классов 11](#_Toc99064225)

[4.4. Типовые процессы в программе 11](#_Toc99064226)

[4.5. Человеко-машинное взаимодействие 11](#_Toc99064227)

[4.6. Реализация ключевых классов 11](#_Toc99064228)

[5. Список использованной литературы и других источников 11](#_Toc99064229)

# Формулировка задания

Правила игры «Шарики»:

* Игра происходит на прямоугольном стакане  NxM заполнен шариками разного цвета.
* Можно взрывать цепочки из шариков одного цвета, находящихся рядом друг с другом. При этом шарики, находящиеся выше, падают вниз.
* Цель игры - оставить на поле как можно меньше шариков.
* Через определённые промежутки  времени порождается новый нижний ряд шариков.

**Подвариант 1 (низкий уровень).**

Вариативность:  Предусмотреть возможность создания новых шариков со специальными возможностями по взрыванию соседних шариков. Шарики должны быть визуально различимы.

Реализовать: шарик, который уничтожает все остальные шарики в ряду.

**Подвариант 2 (низкий уровень).**

Вариативность: предусмотреть различные способы порождения очередного ряда шариков.

Реализовать:

- случайный способ порождения ряда шариков;

- не должны образовываться цепочки длиной более 5 шариков.

# Нефункциональные требования

1. Программа должна быть реализована на языке Java SE 12 с использованием стандартных библиотек, в том числе, библиотеки Swing.
2. Форматирование исходного кода программы должно соответствовать Java Code Conventions, September 12, 1997.

# Первая итерация разработки

### Формулировка упрощённого варианта задания

Правила игры «Шарики»:

* Игра происходит на прямоугольном стакане  NxM заполнен шариками разного цвета.
* Можно взрывать цепочки из шариков одного цвета, находящихся рядом друг с другом. При этом шарики, находящиеся выше, падают вниз.
* Цель игры - оставить на поле как можно меньше шариков.
* Через определённые промежутки  времени порождается новый нижний ряд шариков.

### Функциональные требования (сценарии)

* 1. **Сценарий «Играть»:**

1. По указанию пользователя, Игра стартует.
2. По указанию Игры, Стакан создает Ячейки и формируется из них.
3. По указанию Игры, Стакан заполняет ячейки шариками.
4. По указанию пользователя Шарик выбирается и делает его активным
5. активный шарик создает Последовательность шариков и помещает себя в нее
6. Последовательность ищет в Стакане шарики одного цвета
7. Последовательность шариков сообщает стакану какая цветовая группа образована из выбранного шара
8. Если активный шарик образует группу из трёх и более одноцветных шариков, то они удаляются стаканом
9. Стакан перемещает шарики вниз, которые находятся выше.
10. По указанию Игры, Стакан порождает новый нижний ряд шариков
11. **Сценарий завершается**.
    1. **Дочерний сценарий «**Последовательность ищет в Стакане шарики одного цвета**»**
12. **Пока** не будет достигнут последний шар в последовательности
    1. Проверяется цвет активного шара
    2. Проверяется цвет шара рядом с ним
    3. Если активный шар и шар рядом с ним имеют одинаковый цвет, то последовательность переходит к следующему шару
    4. Иначе сценарий завершается.
13. **Сценарий завершается**.
    1. **Дочерний сценарий** «Если шарик образует группу из трёх и более одноцветных шариков, то они удаляются стаканом»
14. По указанию Стакана, ячейка удаляет находящийся в ней шарик
15. **Сценарий завершается**.
    1. **Дочерний сценарий** «Стакан перемещает шарики вниз, которые находятся выше»
16. Ячейка сообщает, что в ней нет шарика
17. Если в ячейке над ним есть шарик, стакан переносит шарик в пустую ячейку
18. **Сценарий завершается**.
    1. **Дочерний сценарий «**Стакан порождает новый нижний ряд шариков**»**
19. По указанию Игры, формируется новая группа шаров
20. стакан перемещает шары, который находится в нем в следующую ячейку над ним
21. стакан добавляет новую группу шаров в ячейки первого ряда
22. **Сценарий завершается.**
    1. **Альтернативный сценарий – переполнение стакана**
23. Если шарик достигает последней верхней ячейки стакана, и ячейка под ним не пустая
24. То игра выводит сообщение о завершении игры
25. Игра завершается
26. **Сценарий завершается.**

### Словарь предметной области

**Игра** - управляет игровым циклом: инициирует создание всех объектов, определяет окончание игры и оценки пользователя.

**Стакан** – область, состоящая из ячеек.

**Ячейка** – область поля, в которой может находиться шарик.

**Шарик** – объект способный взаимодействовать со стаканом и имеет свет.

**Активный Шарик** – Шарик, выбранный по указанию пользователя

**Последовательность шариков** – Группа шариков от выбранного до края стакана в направлении влево, вправо, вверх и вниз, находит последовательность шариков того же цвета, что и выбранный шарик.

### Структура программы на уровне классов

### Типовые процессы в программе

### Человеко-машинное взаимодействие

### Реализация ключевых классов

# Вторая итерация разработки

### Функциональные требования (сценарии)

### Словарь предметной области

### Структура программы на уровне классов

### Типовые процессы в программе

### Человеко-машинное взаимодействие

### Реализация ключевых классов

# Список использованной литературы и других источников