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 «01» марта 2022г. № 303-ст

2. Срок представления работы к защите « 26 »   мая  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 «04» марта 2022 г.

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

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

«04» марта 2022 г.

**Содержание**

1 Формулировка задания 4

2 Нефункциональные требования 4

3 Первая итерация разработки 5

3.1 Формулировка упрощенного варианта задания 5

3.2 Функциональные требования (сценарии) 5

3.3 Словарь предметной области 10

3.4 Структура программы на уровне классов 11

3.5 Типовые процессы в программе 13

3.6 Человеко-машинное взаимодействие 16

3.7 Реализация ключевых классов 20

3.8 Реализация ключевых тестовых случаев 31

4 Вторя итерация разработки 44

4.1 Функциональные требования (сценарии) 44

4.2 Словарь предметной области 47

4.3 Структура программы на уровне классов 48

4.4 Типовые процессы в программе 50

4.5 Человеко-машинное взаимодействие 52

4.6 Реализация ключевых классов 57

4.7 Реализация ключевых тестовых случаев 69

5 Список использованной литературы и других источников 81

# 1 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.

# 2 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.

# 3 Первая итерация разработки

### 3.1 Формулировка упрощенного 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.

### 3.2 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 последние пять рядов 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. По указанию Игры, Стакан удаляет последовательность, если он образует группу из трёх и более одноцветных шариков
8. По указанию Стакана, Шарики, которые находятся выше удаленной последовательностю, падают вниз
9. По указанию Игры, Через определённые промежутки времени, Стакан порождает новый нижний ряд шариков
10. **Сценарий завершается**.
    1. **Дочерний сценарий «**Последовательность ищет в Стакане шарики одного цвета**»**
11. По указанию Стакана, соседние ячейки ячейки активного шарика проверяются
12. **Для** **всех соседние** **ячейки**
    1. Если шарик в соседней ячейке и активный шарик имеют одинаковый свет, то, соседний шарик создает свою последовательность и помещает себя в нее
    2. Последовательность соседного шарика ищет в Стакане шарики одного цвета
    3. Последовательность соседнего шарика помещаеться в последовательность активного шарика
13. **Сценарий завершается**.
    1. **Дочерний сценарий** «Стакан удаляет последовательность, если он образует группу из трёх и более одноцветных шариков»
14. По указанию Стакана, каждый шарик в последовательности удаляется из своей ячейки
15. Последовательность удаляется из набор шариков в стакане
16. **Сценарий завершается**.
    1. **Дочерний сценарий** «Шарики, которые находятся выше удаленной последовательностю, падают вниз»
17. Ячейка сообщает, что в ней нет шарика
18. Стакан ишет шарик в ячейках выше в том же столбце
19. Cтакан переносит шарик в ячейку, которая отправила сообщение
20. **Сценарий завершается**.
    1. **Дочерний сценарий «**Стакан порождает новый нижний ряд шариков**»**

1. Cтакан перемещает шары, который находится в нем в следующую ячейку над ним

2. Для всех ячеек в последнем ряду ячейки в стакане

2.1. Стакан создает новый шарик случайного цвета и помещает его в ячейку

**3. Сценарий завершается.**

* 1. **Альтернативный сценарий – переполнение стакана**

1. Если шарик достигает последней верхней ячейки стакана, и ячейка под ним не пустая
2. То игра выводит сообщение о завершении игры
3. Игра завершается
4. **Сценарий завершается.**

### 3.3 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 вокруг шарик, находит последовательность шаров того же цвета, что и шарик.

### 3.4 Структура программы на уровне классов

### 3.5 Типовые процессы в программе

### 3.6 Человеко-машинное взаимодействие

### 3.7 Реализация ключевых классов3.8 Реализация ключевых тестовых случаев

# 4 Вторя итерация разработки

### 4.1 Функциональные требования (сценарии)

### 4.2 Словарь предметной области

### 4.3 Структура программы на уровне классов

### 4.4 Типовые процессы в программе

### 4.5 Человеко-машинное взаимодействие

### 4.6 Реализация ключевых классов

### 4.7 Реализация ключевых тестовых случаев

# 5 Список использованной литературы и других источников

**Перечень замечаний к работе**