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 № 27\_01)

Студент: Быстряков Д.В.

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: Быстряков Д.В.

Группа: ПрИн-367

1. Тема: «Проектирование и реализация программы с использованием объектно-ориентированного подхода» (индивидуальное задание – вариант № 27\_01)

Утверждена приказом от «24» января 2022г. № 101-ст

2. Срок представления работы к защите « 04 »   июня  2021 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: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Быстряков Д.В.

«12» февраля 2021 г.

**Содержание**

1 Формулировка задания 4

2 Нефункциональные требования 5

3 Первая итерация разработки 6

3.1 Формулировка упрощенного варианта задания 6

3.2 Функциональные требования (сценарии) 6

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 33

3.7 Реализация ключевых классов 36

3.8 Реализация ключевых тестовых случаев 56

4 Вторая итерация разработки 69

4.1 Функциональные требования (сценарии) 69

4.2 Словарь предметной области 75

4.3 Структура программы на уровне классов 77

4.4 Типовые процессы в программе 80

4.5 Человеко-машинное взаимодействие 84

4.6 Реализация ключевых классов 89

4.7 Реализация ключевых тестовых случаев 114

5 Список использованной литературы и других источников 140

# 1 Формулировка задания

Игра «Запутанная схема»:

* Задача – распутать нитки.

Дополнительные требования:

* предусмотреть в программе **точки расширения**, используя которые можно реализовать  вариативную часть программы (в дополнение к базовой функциональности).

Вариативность:

* предусмотреть появление узлов с различным поведением движения. Узлы должны быть визуально различимы.

Реализовать:

* узел, который не может двигаться;
* узел, который движется только по горизонтали.

# 2 Нефункциональные требования

# 3 Первая итерация разработки

### 3.1 Формулировка упрощённого варианта задания

### 3.2 Функциональные требования (сценарии)

*кому/что поручить*

**1) Сценарий** «Играть»

1. **По указанию пользователя,** Игра стартует.
2. **По указанию** Игры, создается Клубок.
3. **Делать**
   1. **По указанию** **пользователя,** Узел меняет местоположение.
   2. **В ответ на запрос** Игры, Нитки, соединенные с выбранным пользователем Узлом, меняют расположение.
   3. **В ответ на запрос** Игры, Клубок **сообщает** о текущем расположении Ниток и Узлов.
   4. **Если** две и более пересеченных Ниток не пересекаются, то Нитка распутана.

**Пока** имеется в Клубке хотя бы одно пересечение Ниток

1. Игра считает победителем пользователя, который распутал все нитки.
2. **Сценарий завершается**.

**2) Дочерний сценарий «**создается Клубок**»**

1. Клубок создаёт и расставляет Узлы.
2. Клубок создаёт и расставляет от расположения Узла Нитки.
3. **Сценарий завершается**.

**3) Дочерний сценарий** «Узел меняет местоположение»

1. **По указанию** Узла, Клубок даёт разрешение менять расположение.
2. Узел принимает новое расположение.
3. **Сценарий завершается**.

**3.1) Альтернативный сценарий** «Узел перемещается за пределы Клубка». Сценарий **выполняется с** п. 1 сценария 3

1. **В ответ на запрос** Узла, Клубок **сообщает** о невозможности переместить Узел за пределы игрового поля.
2. **Сценарий завершается**.

**3.2) Альтернативный сценарий** «Узел перемещается на место другого Узла». Сценарий **выполняется с** п. 1 сценария 3

1. **В ответ на запрос** Узла, Клубок **сообщает** о невозможности переместить Узел, так как позиция занята другим Узлом.
2. **Сценарий завершается**.

**4) Дочерний сценарий** «Нитки, соединенные с выбранным пользователем Узлом, меняют расположение»

1. **По указанию** Ниток, Клубок даёт разрешение менять расположение.
2. Нитки принимают новое расположение.
3. **Сценарий завершается**.

**5) Альтернативный сценарий** «Досрочное завершение игры». Сценарий **выполняется в любой точке** главного сценария

1. **По указанию пользователя,** программа завершается без определения победителя.
2. **Сценарий завершается.**

### 3.3 Словарь предметной области

*знает/умеет/предназначена*

**Игра** - управляет игровым циклом: инициирует создание всех сущностей, определяет окончание игры и победителя.

**Клубок** – n-угольная область, состоящая из Узлов и Ниток. Знает о расположении Узла и пересекаются Нитки в Клубке или нет.

**Узел** – координатная точка, которую перемещает пользователь, чтобы распутать Клубок. Знает о координатах точки абсциссы X и ординаты Y. Задает позиции так, чтобы каждые две точки образовывали Нитку.

**Нитка** – линия, которая при запуске Игры находится в запутанном состоянии. Растягивается или сужается в зависимости от того, куда пользователь перемещает Узел. Считается распутанной, когда две и более пересеченных Ниток не пересекаются. Знает о координатах начала и конца Узла и длине Нитки.