

**7466**

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
имени В.Ф.УТКИНА**

# **ОФОРМЛЕНИЕ ТЕКСТОВЫХ АЛГОРИТМОВ**

Методические указания  
к лабораторным работам  
и практическим занятиям

Рязань 2022

Оформление текстовых алгоритмов: методические указания к лабораторным работам и практическим занятиям / Рязан. гос. радиотехн. ун-т. им. В.Ф.Уткина; Сост.: В.В. Белов, В.И. Чистякова. Рязань, 2022. 24 с.

Излагаются методические рекомендации по оформлению текстовых алгоритмов при выполнении лабораторных работ и практических заданий по дисциплинам «Проектирование программных систем» и «Проектирование информационных систем». Теоретическую платформу указаний образуют объектно-ориентированная методология проектирования и систематический подход к проектированию, основанный на использовании каскадной модели жизненного цикла программно-информационной системы, принципа ФИСАП и компонентно-ориентированного программирования.

Предназначены для бакалавров всех форм обучения направлений 09.03.03 «Прикладная информатика», 09.03.04 «Программная инженерия» и всех желающих изучить вопросы проектирования программно-информационных систем.

Библиогр.: 6 назв.

*Жизненный цикл, модель, анализ, проектирование, реализация, UML, архитектура, поведение, класс, сценарий, автомат, компонент.*

Рецензент: кафедра вычислительной и прикладной математики Рязанского государственного радиотехнического университета (зав. кафедрой д-р техн. наук, проф. Г.В. Овечкин)

## Проектирование программно-информационных систем

Составители   Белов Владимир Викторович  
                          Чистякова Валентина Ивановна

Подписано в печать XX.XX.2022. Усл. печ. л. 1,5.

Тираж 1 экз.

Рязанский государственный радиотехнический университет  
390005, Рязань, ул. Гагарина, 59/1.  
Редакционно-издательский центр РГРТУ.

## СОДЕРЖАНИЕ

<b>1</b>	<b>ДИАГРАММЫ ПРЕЦЕДЕНТОВ.....</b>	<b>4</b>
1.1	О спецификации диаграммы прецедентов .....	4
1.1.1	Назначение спецификации диаграммы прецедентов .....	4
1.1.2	Элементы спецификации диаграммы прецедентов.....	4
1.2	Пример документирования диаграммы.....	5
1.2.1	Диаграмма программной системы решения алгебраического уравнения порядка не выше второго .....	5
1.2.2	Спецификация диаграммы прецедентов .....	6
<b>2</b>	<b>ДИАГРАММЫ ДЕЯТЕЛЬНОСТИ.....</b>	<b>10</b>
2.1	Общие замечания .....	10
2.2	Диаграмма деятельности для прецедента «Решение уравнения».....	10
2.3	Диаграмма деятельности для прецедента «Анализ вида решения» .....	11
2.4	Диаграмма деятельности для прецедента «Вычисление корней линейного уравнения» .....	11
2.5	Диаграмма деятельности для прецедента «Вычисление корней квадратного уравнения» .....	12
2.6	Диаграмма деятельности для прецедента «Вывод результатов».....	12
<b>3</b>	<b>ДИАГРАММЫ ВЗАИМОДЕЙСТВИЯ .....</b>	<b>13</b>
3.1	Диаграмма коммуникации для предметной области «Алгебраическое уравнение» .....	13
3.2	Диаграмма последовательности для предметной области «Алгебраическое уравнение» .....	13
3.3	Отношение между диаграммами коммуникации и последовательности .....	13
<b>4</b>	<b>СПЕЦИФИКАЦИИ ПОТОКОВ – ТЕКСТОВЫЕ АЛГОРИТМЫ (СЦЕНАРИИ) РЕАЛИЗАЦИИ ПРЕЦЕДЕНТОВ.....</b>	<b>14</b>
4.1	Предварительные замечания .....	14
4.2	Сценарий решения уравнения.....	16
	<b>ПРИЛОЖЕНИЕ А .....</b>	<b>22</b>
	<b>БИБЛИОГРАФИЧЕСКИЙ СПИСОК .....</b>	<b>24</b>

# 1 ДИАГРАММЫ ПРЕЦЕДЕНТОВ

## 1.1 О спецификации диаграммы прецедентов

### 1.1.1 Назначение спецификации диаграммы прецедентов

Спецификация позволяет:

- 1) пояснить функциональность прецедентов, представленную на диаграмме одним названием; одного названия часто бывает недостаточно;
- 2) обосновать принятые решения относительно отношений, представленных в виде линий на диаграмме;
- 3) модельеру лучше проникнуть в суть отображаемых элементов диаграммы;
- 4) читателю получить при необходимости дополнительную информацию о сути того, что представлено на диаграмме.

### 1.1.2 Элементы спецификации диаграммы прецедентов

Спецификация диаграммы прецедентов имеет две составляющие:

- 1) характеристика диаграммы прецедентов – документирование элементов диаграммы;
- 2) спецификация потоков – текстовые представления алгоритмов реализации прецедентов, представленных на диаграмме прецедентов.

Характеристика диаграммы прецедентов должна содержать следующие части.

1. Количественная характеристика диаграммы.
2. Характеристика акторов.
3. Характеристика прецедентов.
4. Характеристика отношений.

Для каждого прецедента указывается: название и краткое изложение функциональности. Для каждой дуги указывается и обосновывается отображаемое отношение.

Для демонстрации структурных составляющих приведём в качестве примера фрагмент характеристики некоторой *абстрактной* диаграммы прецедентов.

#### ***Количественная характеристика диаграммы***

Диаграмма содержит:

- акторов: 1;
- прецедентов: 27;
- дуг (отношений): 16

#### ***Характеристика акторов:***

Актор с ролью «Аналитик» представляет пользователя системы.

#### ***Характеристика прецедентов:***

---

Белов В.В., Чистякова В.И. Проектирование программно-информационных систем. Методические указания к лабораторным работам и практическим занятиям

Ведение базы данных (БД) – базисный прецедент, связанный с актором обязательной ассоциацией. Одна из основных функций системы, обеспечивающих достижение цели создания системы.

Ведение БД «Задачи» – вспомогательный прецедент, отражающий одну из подзадач базисного прецедента Ведение базы данных (БД) и связанный с ним отношением зависимости вида расширение.

### ***Характеристика отношений***

Актор находится в отношении ассоциации со своими базисными прецедентами. Все ассоциации являются обязательными, потому что полноценное использование системы предполагает обязательное использование функциональности каждого прецедента.

Все отношения между прецедентами являются зависимостями вида расширение, потому что при реализации зависящих прецедентов использование вспомогательной функциональности не является обязательным, а определяется конкретикой текущих условий.

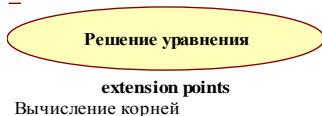
Конкретный сценарий реализации функциональности прецедента Ведение базы данных (БД) обычно реализует только одну из подзадач: 1) ведение БД «Задачи»; 2) ведение БД «Решаемые задачи»; 3) ведение БД «Решённые (архивные) задачи».

Конкретный сценарий реализации функциональности прецедента Отображение вербальных значений в числа обычно реализует только одну из подзадач: 1) отображение А-слов в числа; 2) отображение В-слов в числа; 3) отображение С-слов в числа.

## **1.2 Пример документирования диаграммы**

### **1.2.1 Диаграмма программной системы решения алгебраического уравнения порядка не выше второго**

Диаграмма прецедентов отображает функциональность проектируемой системы в виде совокупности прецедентов и отношений между ними. Пример диаграммы представлен на рисунке 1. На диаграмме представлен один актор – конкретная роль потребителя функциональности системы. Заметим, что на рисунке 1 использована форма представления базисного прецедента в нотации Rational Rose. В StarUML форма представления прецедентов с точками ветвления иная:



Стрелки зависимости типа extend должны сопровождаться обязательным указанием условия применения вспомогательного прецедента.

Прагматика UML рекомендует обязательное указание ещё точки входа в базисный прецедент. Условие применения вспомогательного прецедента обязательно должно быть представлено в свойстве Condition стрелки отношения. Это условие вместе с точкой входа следует разместить ещё и в дополнительном пояснении, размещаемом рядом со стереотипом «extend». Стандарт UML [5] предлагает оформлять это пояснение с помощью инструмента Note в виде замечания. При этом условие записывается как ограничение (текст в фигурных скобках) с использованием Constraint – механизма одного из расширений UML и предваряется словом Condition. Точка входа записывается в следующей строке за ограничением и предваряется словосочетанием extension point. Пример такого оформления пояснения имеет вид:

Constraint: {вычисление корней}  
extension point: уравнение квадратное

В то же время Стандарт UML разрешает для оформления пояснений extend-зависимостей применение нотаций, предлагаемых поставщиками систем UML-моделирования. Мы рекомендуем применять нотацию системы моделирования Rational Rose от фирмы Rational Software Corporation. Следуя указанной нотации, пояснения extend-зависимостей следует оформлять в виде текста, создаваемого с помощью инструмента Text. При этом условие записывается в квадратных, а не фигурных скобках. Точка входа записывается в круглых скобках. Причём точка входа записывается первой. Пример оформления пояснения extend-зависимостей в нотации Rational Rose показан на рисунке 1.

### 1.2.2 Спецификация диаграммы прецедентов

#### *Количественная характеристика диаграммы*

Диаграмма содержит:

- акторов: 1;
- прецедентов: 5;
- дуг (отношений): 5

#### *Характеристика акторов*

Актор с ролью «Студент» представляет пользователя системы. Он использует единственную значимую для него функцию системы – «Решение уравнения». Информация, ассоциированная с актором, представляет собой значения коэффициентов уравнения и результат решения уравнения. Указанная информация хранится в сознании актора, внутренних переменных программы и внешних носителях, если ввод или вывод осуществляется с использованием этих носителей.

#### *Характеристика прецедентов*

Белов В.В., Чистякова В.И. Проектирование программно-информационных систем. Методические указания к лабораторным работам и практическим занятиям

Решение уравнения – базисный прецедент, связанный с актором обязательной ассоциацией. Это единственная основная функция системы, доставляющая актору значимый для него результат. Остальные прецеденты – Анализ вида решения, Вывод результатов, Вычисление корней линейного уравнения, вычисление корней квадратного уравнения – являются вспомогательными, их назначение – реализация функциональности базисного прецедента Решение уравнения.

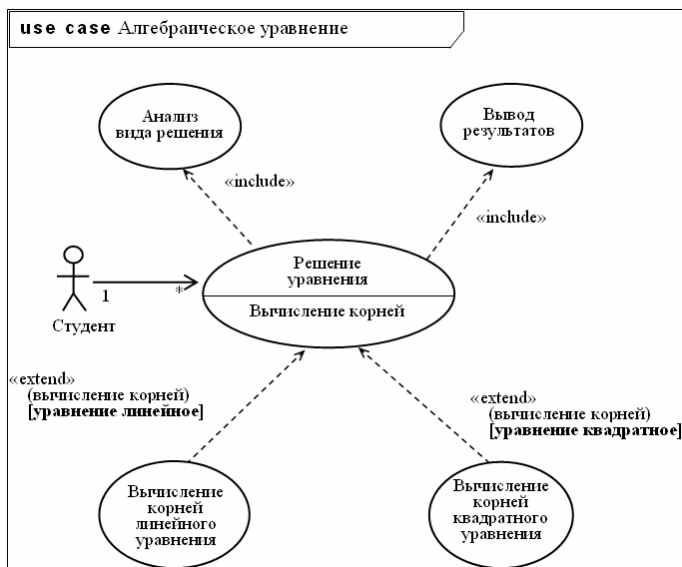


Рисунок 1 – Функции системы для решения алгебраического уравнения

Анализ вида решения – прецедент, представляющий собой обязательную часть функциональности Решение уравнения. Это функция состоит в выяснении порядка решаемого алгебраического уравнения и выяснения количества его корней. Порядок уравнения по заданию может иметь значения: 0, 1, 2.

Обратим внимание на то, что решается не квадратное уравнение, а произвольное алгебраическое уравнение с порядком не выше второго, т. е. условие  $a \neq 0$  не накладывается. Заметим, что задаваемое уравнение может и не иметь корней, – если уравнение имеет нулевой порядок, а свободный член отличен от нуля. Этот случай предполагает получение решения на этапе анализа уравнения. Решением при этом является установление факта отсутствия корней.

Вычисление корней линейного уравнения имеет тривиальную реализацию, поэтому не уточняется.

Вычисление корней квадратного уравнения – реализуется стандартно. Особенность реализации – результат решения представляется в виде строки с характеристикой корней и их значениями.

Вывод результатов – это обычный вывод текстовой строки, сформированной прецедентами анализа и решения уравнения.

### Характеристика отношений

Отношение между актором Студент и прецедентом Решение уравнения является ассоциацией. Это отношение представлено на диаграмме линией с указанными направлением отношения и мощностью (кратностью) ассоциации.

Ассоциация выбрана направленной, потому что актор является пассивным участником взаимодействия с системой – инициирует выполнение прецедента, задаёт исходные данные, ожидает решения и получает результат. Диалога прецедента с актором нет.

Элементы кратности отношения 1 – \* означают следующее:

«1» – каждое конкретное уравнение решается единственным актором **И** для выполнения прецедента требуется инициирование (запуск) со стороны актора; это означает, что участие актора в рассматриваемой ассоциации является обязательным;

\* (эквивалентно 0..\*) – один экземпляр актора, т. е. один конкретный актор может решать несколько уравнений **И** конкретный актор может вовсе не решать никакое уравнение; это означает, что участие прецедента в рассматриваемой ассоциации является необязательным.

Отношение между базисным прецедентом Решение уравнения и прецедентом Анализ вида решения является разновидностью отношения зависимости, называемой включением, что символизируется стереотипом «include». Это отношение отражает тот факт, что функциональность прецедента Анализ вида решения является обязательной составляющей прецедента Решение уравнения, т. е. функциональность Анализ вида решения безусловно и всегда реализуется при реализации функциональности Решение уравнения.

Отношение между базисным прецедентом Решение уравнения и прецедентом Вывод результата так же является разновидностью отношения зависимости, называемой включением, что и символизируется стереотипом «include». Это отношение отражает тот факт, что функциональность прецедента Вывод результата является обязательной составляющей прецедента Решение уравнения, т. е. функциональность Вывод результата безусловно и всегда реализуется при реализации функциональности Решение уравнения.

---

Белов В.В., Чистякова В.И. Проектирование программно-информационных систем. Методические указания к лабораторным работам и практическим занятиям



Отношение между базисным прецедентом Решение уравнения и прецедентом Вычисление корней линейного уравнения является разновидностью отношения зависимости, называемой расширением, что и символизируется стереотипом «extended». Это отношение отражает тот факт, что функциональность прецедента Вычисление корней линейного уравнения является необязательной составляющей прецедента Решение уравнения: функциональность Вычисление корней линейного уравнения реализуется только тогда, когда решаемое алгебраическое уравнение оказывается линейным.

Отношение между базисным прецедентом Решение уравнения и прецедентом Вычисление корней квадратного уравнения является разновидностью отношения зависимости, называемой расширением, что и символизируется стереотипом «extended». Это отношение отражает тот факт, что функциональность прецедента Вычисление корней квадратного уравнения является необязательной составляющей прецедента Решение уравнения: функциональность Вычисление корней квадратного уравнения реализуется только тогда, когда решаемое алгебраическое уравнение оказывается квадратным.

#### Способ реализации документирования диаграммы прецедентов

Документированию подлежат все элементы всех диаграмм. Оформлять документирование (характеристику диаграммы) следует следующим образом: первоначально необходимый текст характеристики создаётся в текстовом процессоре Word – это позволит устранить возможные орфографические ошибки с помощью Spelling Checker’a текстового процессора, а затем части характеристики разносятся по соответствующим секциям Documentation диаграммы. Текстовый документ характеристики помещается в текст отчёта по изучаемой теме вместе с тестами алгоритмов реализации прецедентов.

Тесты алгоритмов реализации прецедентов рекомендуется оформлять в виде таблиц (стандарт фирмы EPAM Systems). Пример часто применяемого безтабличного оформления тестовых алгоритмов приведён в приложении А.

Следует не забывать, что документированию подлежат не только элементы диаграмм, но сами диаграммы. Документирование диаграммы осуществляется заполнением секции Documentation в то время, когда не активирован ни один из элементов этой диаграммы.

## 2 ДИАГРАММЫ ДЕЯТЕЛЬНОСТИ

### 2.1 Общие замечания

Диаграммы деятельности используются для представления реализаций (алгоритмов) прецедентов, представленных на диаграмме прецедентов. Поэтому количество диаграмм деятельности должно совпадать с количеством прецедентов. Все эти диаграммы деятельности должны иметь спецификацию. Эти спецификации представляют собой спецификации соответствующих прецедентов.

### 2.2 Диаграмма деятельности для прецедента «Решение уравнения»

Диаграмма представлена на рисунке 2.

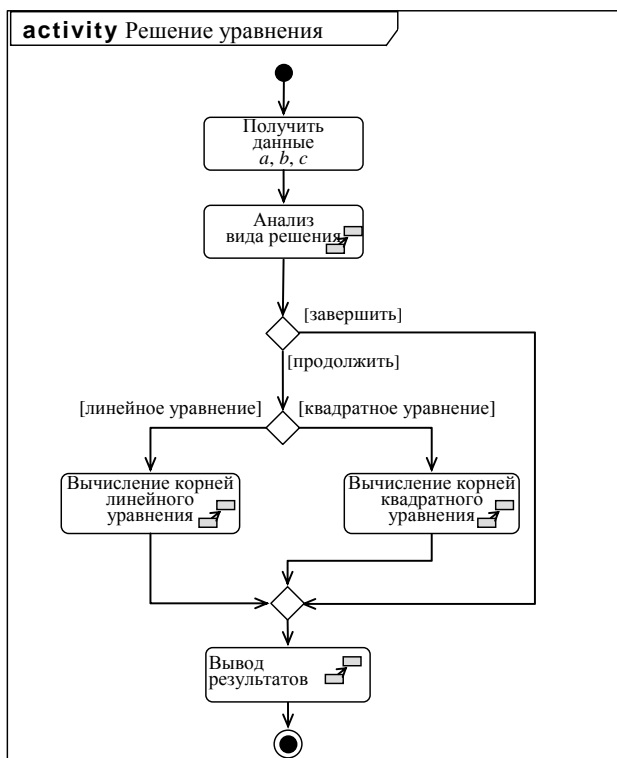


Рисунок 2 – Диаграмма деятельности для прецедента «Решение уравнения»

## 2.3 Диаграмма деятельности для прецедента «Анализ вида решения»

Диаграмма представлена на рисунке 3.

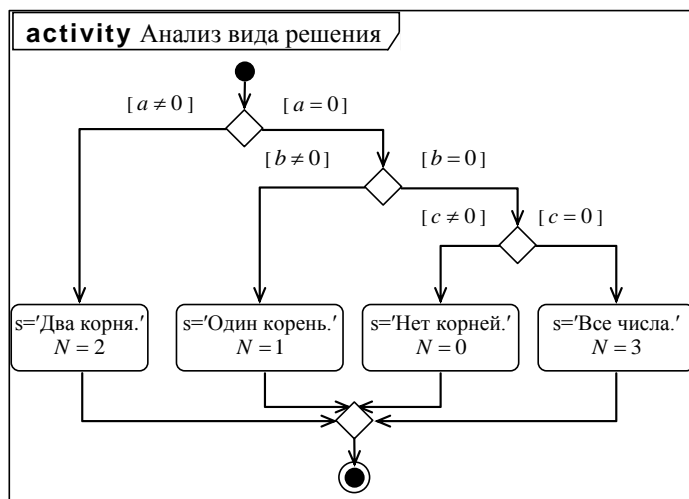


Рисунок 3 – Диаграмма деятельности для прецедента  
«Анализ вида решения»

## 2.4 Диаграмма деятельности для прецедента «Вычисление корней линейного уравнения»

Диаграмма представлена на рисунке 4.

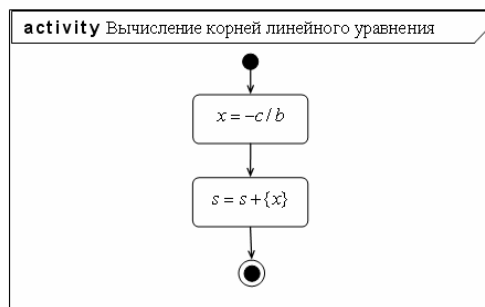


Рисунок 4 – Диаграмма деятельности для прецедента  
«Вычисление корней линейного уравнения»

## 2.5 Диаграмма деятельности для прецедента «Вычисление корней квадратного уравнения»

Диаграмма представлена на рисунке 5.

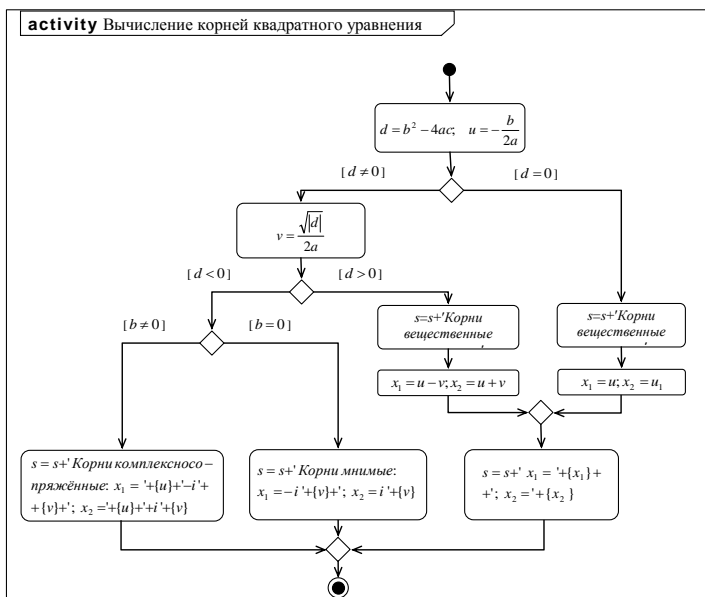


Рисунок 5 – Диаграмма деятельности для прецедента  
«Вычисление корней квадратного уравнения»

## 2.6 Диаграмма деятельности для прецедента «Вывод результатов»

Диаграмма представлена на рисунке 6.

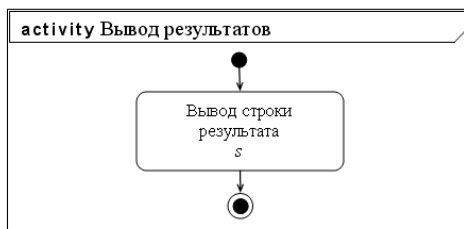


Рисунок 6 – Диаграмма деятельности для прецедента  
«Вывод результатов»

### 3 ДИАГРАММЫ ВЗАИМОДЕЙСТВИЯ

#### 3.1 Диаграмма коммуникации для предметной области «Алгебраическое уравнение»

Диаграмма представлена на рисунке 7.

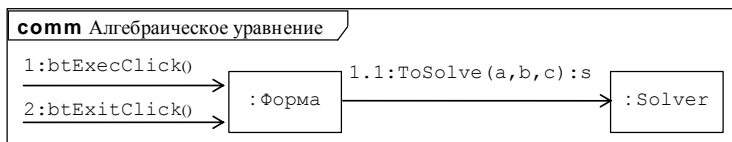


Рисунок 7 – Диаграмма коммуникации для предметной области «Алгебраическое уравнение»

#### 3.2 Диаграмма последовательности для предметной области «Алгебраическое уравнение»

Диаграмма представлена на рисунке 8.

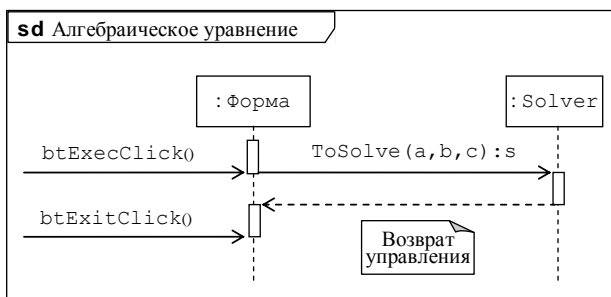


Рисунок 8 – Диаграмма последовательности для предметной области «Алгебраические уравнение»

#### 3.3 Отношение между диаграммами коммуникации и последовательности

Диаграммы коммуникации и последовательности взаимобразно связаны отношением derive. Это означает, что каждая из этих диаграмм может быть получена из другой диаграммы. Обе диаграммы содержат одну и ту же информацию (один и тот же контент), но в разной графической нотации. На диаграммах представлены роли двух классов – «Форма» и «Solver». Сообщения, показанные над стрелками, представляют собой обращения к методам этих классов. Эти методы (операции) должны обязательно присутствовать на соответствующей диаграмме классов.

## **4 СПЕЦИФИКАЦИИ ПОТОКОВ – ТЕКСТОВЫЕ АЛГОРИТМЫ (СЦЕНАРИИ) РЕАЛИЗАЦИИ ПРЕЦЕДЕНТОВ**

### **4.1 Предварительные замечания**

Спецификации представлены в таблицах 1 –5. Из пяти прецедентов, представленных на рисунке 1, описаны только три (для образца этого достаточно, идея построения спецификации просматривается достаточно ясно). Для прецедента «Анализ вида решения» представлено три варианта спецификации. В реальной практике следует использовать вариант, предусмотренный внутрикорпоративными стандартами и инструкциями.

Варианты спецификации отличаются, главным образом, способом представления альтернативных потоков, включая вложенные альтернативы (альтернативные потоки альтернативных потоков). Имеет место один общий принцип: строки с отступом (предпочтительно) или выступлениями представляют собой реализации потока, относительно заголовка которого они сдвинуты. Использование многоуровневой нумерации строк для указания их «владельца» пригодны только при малом уровне ветвления, в противном случае они теряют читаемость.

Заметим, что потоки, альтернативные альтернативным потоком размещаются в одной секции альтернативных потоков. Конечно же, можно было бы не выделять особо основной поток и записывать весь алгоритм по принципу записи альтернативных потоков, но такое выделение помогает во многих случаях представить основной поток в наиболее удобочитаемом виде.

Следует понимать, что ветвление алгоритма реализуется либо с помощью пунктов потока типа «Выполняется проверка условия», либо с помощью ссылок на альтернативы с последующим размещением определения (описания) альтернативного потока в специальной секции таблицы. Всё как в программировании: в тексте программы (аналог основного потока) имеются ссылки на подпрограммы (альтернативные потоки) и имеются места, где содержатся определения – тексты подпрограмм.

Список таблиц:

Таблица 1 – Описание прецедента «Решение уравнения»

Таблица 2 – Описание прецедента «Анализ вида решения». Вариант № 1

Таблица 3 – Описание прецедента «Анализ вида решения». Вариант № 2

Таблица 4 – Описание прецедента «Анализ вида решения». Вариант № 3

Таблица 5 – Описание прецедента «Вычисление корней квадратного уравнения».

Таблица 6 – Описание прецедента «Анализ вида решения». Вариант № 4

---

Белов В.В., Чистякова В.И. Проектирование программно-информационных систем. Методические указания к лабораторным работам и практическим занятиям

В таблице 2 представлен подход к построению описания алгоритма, согласно которому в основном потоке точки ветвления представляются пунктами аналогичными оператору switch языка C, – например, (Решение по коэффициенту  $a$ ), а условия ветвлений (аналоги строк case) и соответствующие им действия размещаются в секции описания альтернативных потоков.

В таблице 3 обращения к альтернативам так же представляются обычными пунктами основного потока с явным указанием проверяемых условий, как в сокращённой форме оператора if. Пункты основного потока располагаются в последовательности, определяемой справедливостью одного из условий. Этот принцип иногда называют движением по ветви «Да». Его удобно применять в случае ветвлений на два направления.

В таблице 4 и 6 представлен тот же подход, что и в таблице 3, но в неудобочитаемой форме – в основном потоке условия ветвления не упоминаются вовсе, хотя, конечно же, их можно представить в виде комментариев. Вместо этого считается, что потенциально каждый из пунктов основного потока может иметь альтернативу, поэтому перед выполнением пункта основного потока следует посмотреть в секцию альтернативных потоков и выяснить – нет ли у этого пункта альтернативы. Альтернативный участок маркируется номером пункта основного потока, дополненным буквой, и условием выполнения альтернативы в квадратных скобках. Если условие оказывается справедливым, то выполнение алгоритма продолжается по альтернативной ветви. Тот принцип используется и при описании альтернатив альтернативам.

В таблице 5 представлен подход, пожалуй, наиболее читабельный. В нём ссылки на альтернативные потоки и потоки ошибок представляются в основном потоке в виде пунктов с особыми идентификаторами, начинающимися с букв A, E или S с последующей записью условия ветвления в квадратных скобках. Буквы символизируют следующее: A – альтернативный поток, E – поток ошибок, S – подпоток, – используемый для алгоритмической реализации вспомогательных прецедентов и оформляемый в отдельной таблице

Определение альтернатив начинается теми же идентификаторами, что и ссылки на альтернативы, но им не предшествует номер пункта в потоке. Например, если пункт со ссылкой имеет вид 2. A1 [ $a \neq 0$ ], то определение альтернативы начинается с заголовка A1 [ $a \neq 0$ ].

## 4.2 Сценарий решения уравнения

Таблица 1– Описание прецедента «Решение уравнения»

Название:	Решение уравнения
Действующие лица:	Пользователь
Краткое описание:	Решает уравнение $ax^2 + bx + c = 0$ без ограничений на коэффициенты
Предусловия:	У пользователя возникла потребность решить уравнение. Он запустил Систему
Постусловия:	Система отображает решение
Основной поток (нормальное течение):	<ol style="list-style-type: none"> <li>1. Система отображает окно с элементами для ввода коэффициентов <math>a, b, c</math></li> <li>2. Система получает признак завершения набора данных пользователем</li> <li>3. S1: Анализ вида решения // Устанавливается значение переменной <math>N</math></li> <li>4. S2. [ <math>N = 2</math> ]: Вычисление корней квадратного уравнения</li> <li>5. S3. [ <math>N = 1</math> ]: Вычисление корней линейного уравнения</li> <li>6. S4: Вывод результатов</li> <li>7. Выполнение прецедента завершается</li> </ol>
Альтернативный поток (альтернативные течения):	// В описании основного потока нет ссылок на альтернативные потоки. Отсутствуют ссылки и на потоки ошибок. Имеются ссылки только на подпотоки, но они представлены отдельными таблицами.
Приоритет (Критично   Важно   Желательно):	Критично
Частота использования (Всегда   Часто   Иногда   Редко   Один раз):	Всегда



Таблица 2 – Описание прецедента «Анализ вида решения». Вариант № 1

Название:	S1. Анализ вида решения
Действующие лица:	Пользователь
Краткое описание:	Выясняется количество корней уравнения.
Предусловия:	Пользователь завершил ввод исходных данных – значений коэффициентов $a, b, c$
Постусловия:	Система получает признак количества решений уравнения и начальную часть строки вывода
Основной поток (нормальное течение):	<ol style="list-style-type: none"> <li>1. Присваивается <math>s = \text{'Вид решения уравнения.'}</math></li> <li>2. (Решение по коэффициенту <math>a</math>)</li> <li>3. Выполнение прецедента завершается</li> </ol>
Альтернативный поток (альтернативные течения):	<ol style="list-style-type: none"> <li>2a. [Получено <math>a \neq 0</math> ] <ol style="list-style-type: none"> <li>2a1. Присваивается <math>s = s + \text{'два числа'}</math></li> <li>2a2. Присваивается <math>N = 2</math></li> <li>2a3. Переход на п. 3 основного потока</li> </ol> </li> <li>2b. [Получено <math>a = 0</math> ] <ol style="list-style-type: none"> <li>2b1. (Решение по коэффициенту <math>b</math> ) <ol style="list-style-type: none"> <li>2b1a. [Получено <math>b \neq 0</math> ] <ol style="list-style-type: none"> <li>2b1a1. Присваивается <math>s = s + \text{'одно число'}</math></li> <li>2b1a2. Присваивается <math>N = 1</math></li> <li>2b1a3. Переход на п. 3 основного потока</li> </ol> </li> <li>2b1b. [Получено <math>b = 0</math> ] <ol style="list-style-type: none"> <li>2b1a1. (Решение по коэффициенту <math>c</math> ) <ol style="list-style-type: none"> <li>2b1a1a. [Получено <math>c \neq 0</math> ] <ol style="list-style-type: none"> <li>2b1a1a1. Присваивается <math>s = s + \text{'нет решений'}</math></li> <li>2b1a1a2. Присваивается <math>N = 0</math></li> <li>2b1a1a3. Переход на п. 3 основного потока</li> </ol> </li> <li>2b1a1b. [Получено <math>c = 0</math> ] <ol style="list-style-type: none"> <li>2b1a1b1. Присваивается <math>s = s + \text{'все числа'}</math></li> <li>2b1a1b2. Присваивается <math>N = 3</math></li> <li>2b1a1b3. Переход на п. 3 основного потока</li> </ol> </li> </ol> </li> </ol> </li> </ol> </li></ol></li></ol>
Приоритет (Критично   Важно   Желательно):	Критично
Частота использования (Всегда   Часто   Иногда   Редко   Один раз):	Всегда

Таблица 3 – Описание прецедента «Анализ вида решения». Вариант № 2

Название:	S1. Анализ вида решения
Действующие лица:	Система
Краткое описание:	Выясняется количество корней уравнения.
Предусловия:	Пользователь завершил ввод исходных данных – значений коэффициентов $a, b, c$
Постусловия:	Система получает признак количества решений уравнения и начальную часть строки вывода
Основной поток (нормальное течение):	<ol style="list-style-type: none"> <li>1. Присваивается <math>s = \text{'Вид решения уравнения.'}</math></li> <li>2. Выполняется проверка условия <math>a = 0</math></li> <li>3. Выполняется проверка условия <math>b = 0</math> // [Получено <math>a = 0</math>]</li> <li>4. Выполняется проверка условия <math>c = 0</math> // [Получено <math>b = 0</math>]</li> <li>5. Присваивается <math>s = s + \text{'все числа'}</math> // [Получено <math>c = 0</math>]</li> <li>6. Присваивается <math>N = 3</math></li> <li>7. Выполнение прецедента завершается</li> </ol>
Альтернативный поток (альтернативные течения):	<ol style="list-style-type: none"> <li>3a. [Получено <math>a \neq 0</math>] 3a1. Присваивается <math>s = s + \text{'два числа'}</math> 3a2. Присваивается <math>N = 2</math> 3a3. Переход на п. 7 основного потока</li> <li>4a. [Получено <math>b \neq 0</math>] 4a1. Присваивается <math>s = s + \text{'одно число'}</math> 4a2. Присваивается <math>N = 1</math> 4a3. Переход на п. 7 основного потока</li> <li>5a. [Получено <math>c \neq 0</math>] 5a1. Присваивается <math>s = s + \text{'нет решений'}</math> 5a2. Присваивается <math>N = 0</math> 5a3. Переход на п. 7 основного потока</li> </ol>
Приоритет (Критично   Важно   Желательно):	Критично
Частота использования (Всегда   Часто   Иногда   Редко   Один раз):	Всегда

Таблица 4 – Описание прецедента «Анализ вида решения». Вариант № 3

Название:	S1. Анализ вида решения
Действующие лица:	Система
Краткое описание:	Выясняется количество корней уравнения.
Предусловия:	Пользователь завершил ввод исходных данных – значений коэффициентов $a, b, c$
Постусловия:	Система получает признак количества решений уравнения и начальную часть строки вывода
Основной поток (нормальное течение):	<ol style="list-style-type: none"> <li>1. Присваивается <math>s = \text{'Вид решения уравнения.'}</math></li> <li>2. Присваивается <math>s = s + \text{'два числа,'}</math></li> <li>3. Присваивается <math>N = 2</math></li> <li>4. Выполнение прецедента завершается</li> </ol>
Альтернативный поток (альтернативные течения):	<ol style="list-style-type: none"> <li>2a. [Получено <math>a = 0</math> ] <ol style="list-style-type: none"> <li>2a1. Присваивается <math>s = s + \text{'одно число,'}</math></li> <li>2a2. Присваивается <math>N = 1</math></li> <li>2a3. Переход на п. 4 основного потока</li> </ol> </li> <li>2a1a. [Получено <math>b = 0</math> ] <ol style="list-style-type: none"> <li>2a1a1. Присваивается <math>s = s + \text{'нет решений'}</math></li> <li>2a1a2. Присваивается <math>N = 0</math></li> <li>2a1a3. Переход на п. 4 основного потока</li> </ol> </li> <li>2a1a1a. [Получено <math>c = 0</math> ] <ol style="list-style-type: none"> <li>2a1a1a1. Присваивается <math>s = s + \text{'все числа.'}</math></li> <li>2a1a1a2. Присваивается <math>N = 3</math></li> <li>2a1a1a3. Переход на п. 4 основного потока</li> </ol> </li> </ol>
Приоритет (Критично   Важно   Желательно):	Критично
Частота использования (Всегда   Часто   Иногда   Редко   Один раз):	Всегда

Таблица 5 – Описание прецедента «Анализ вида решения». Вариант № 4

Название:	S1. Анализ вида решения
Действующие лица:	Система
Краткое описание:	Выясняется количество корней уравнения.
Предусловия:	Пользователь завершил ввод исходных данных – значений коэффициентов $a, b, c$
Постусловия:	Система получает признак количества решений уравнения и начальную часть строки вывода
Основной поток (нормальное течение):	<ol style="list-style-type: none"> <li>1. Присваивается <math>s = \text{'Вид решения уравнения.'}</math></li> <li>2. A1 [ <math>a \neq 0</math> ]</li> <li>3. A2 [ <math>a = 0</math> ]</li> <li>4. Выполнение прецедента завершается</li> </ol>
Альтернативный поток (альтернативные течения):	<p>A1. [ <math>a \neq 0</math> ]</p> <ol style="list-style-type: none"> <li>1. Присваивается <math>s = s + \text{' Два корня, '}</math></li> <li>2. Присваивается <math>N = 2</math></li> <li>3. Переход на пункт 4 основного потока</li> </ol> <p>A2. [ <math>a = 0</math> ]</p> <ol style="list-style-type: none"> <li>1. A3. [ <math>b \neq 0</math> ] <ol style="list-style-type: none"> <li>1. Присваивается <math>s = s + \text{' Один корень, '}</math></li> <li>2. Присваивается <math>N = 1</math></li> <li>3. Переход на пункт 4 основного потока</li> </ol> </li> <li>2. A4. [ <math>b = 0</math> ] <ol style="list-style-type: none"> <li>1. A5. [ <math>c \neq 0</math> ] <ol style="list-style-type: none"> <li>1. Присваивается <math>s = s + \text{' Нет корней, '}</math></li> <li>2. Присваивается <math>N = 0</math></li> <li>3. Переход на пункт 4 основного потока</li> </ol> </li> <li>2. A6. [ <math>c = 0</math> ] <ol style="list-style-type: none"> <li>1. Присваивается <math>s = s + \text{' все числа. '}</math></li> <li>2. Присваивается <math>N = 3</math></li> <li>3. Переход на п. 4 основного потока</li> </ol> </li> </ol> </li> </ol>
Приоритет (Критично   Важно   Желательно):	Критично
Частота использования (Всегда   Часто   Иногда   Редко   Один раз):	Всегда

Таблица 6 – Описание прецедента «Вычисление корней квадратного уравнения»

Название:	S2. Вычисление корней квадратного уравнения
Действующие лица:	Система
Краткое описание:	Определение корней квадратного уравнения.
Предусловия:	Пользователь завершил ввод исходных данных – значений коэффициентов $a, b, c$ . Система выполнила анализ уравнения и установила, что уравнения является квадратным
Постусловия:	Система вычислила корни квадратного уравнения и закончила формирование строки результата.
Основной поток (нормальное течение):	<ol style="list-style-type: none"> <li>1. Присваивается <math>d = b^2 - 4ac</math></li> <li>2. Присваивается <math>u = -b/(2a)</math></li> <li>3. Присваивается <math>s = s + \text{'корни вещественные и равные.'}</math></li> <li>4. Присваивается <math>x_1 = u</math></li> <li>5. Присваивается <math>x_2 = u</math></li> <li>6. Присваивается <math>s = s + \text{' } x_1 = \{x_1\} + \text{'}; x_2 = \{x_2\}</math></li> <li>7. Выполнение прецедента завершается</li> </ol>
Альтернативный поток (альтернативные течения):	<ol style="list-style-type: none"> <li>3а. [Получено <math>d \neq 0</math>]</li> <li>3а1. Присваивается <math>v = \sqrt{ d }/(2a)</math></li> <li>3а2. Присваивается <math>s = s + \text{'корни вещественные различные.'}</math></li> <li>3а3. Присваивается <math>x_1 = u - v</math></li> <li>3а4. Присваивается <math>x_2 = u + v</math></li> <li>3а5. Переход на пункт 6 основного потока</li> <li>3а2а. [Получено <math>d &lt; 0</math>]</li> <li>3а2а1. Присваивается <math>s = s + \text{'корни мнимые:'}</math>  <math>x_1 = \text{' } + -i + \{v\} + \text{'}; x_2 = \text{' } + i + \{v\}</math></li> <li>3а2а3. Переход на пункт 7 основного потока</li> <li>3а2а1а. [Получено <math>b \neq 0</math>]</li> <li>3а2а1. Присваивается <math>s = s + \text{'корни комплексносопряжённые:'}</math>  <math>x_1 = \text{' } + \{u\} + -i + \{v\} + \text{'}; x_2 = \text{' } + \{u\} + i + \{v\}</math></li> <li>3а2а2. Переход на пункт 7 основного потока</li> </ol>
Приоритет (Критично   Важно   Желательно):	Критично
Частота использования (Всегда   Часто   Иногда   Редко   Один раз):	Всегда

## ПРИЛОЖЕНИЕ А

### Оформление потока реализации прецедента «Покупка авиабилета» в виде простого текста

**Предусловие:** перед началом этого элемента Use Case должен быть выполнен элемент Use Case «Заполнить базу данных авиарейсов».

#### Главный поток

Этот элемент Use Case начинается, когда покупатель регистрируется в системе и вводит свой пароль. Система проверяет, правилен ли пароль (E-1), и предлагает покупателю выбрать одно из действий: СОЗДАТЬ, УДАЛИТЬ, ПРОВЕРИТЬ, ВЫПОЛНИТЬ, ВЫХОД.

1. Если выбрано действие СОЗДАТЬ, выполняется подпоток S-1: создать заказ авиабилета.

2. Если выбрано действие УДАЛИТЬ, выполняется подпоток S-2: удалить заказ авиабилета.

3. Если выбрано действие ПРОВЕРИТЬ, выполняется подпоток S-3: проверить заказ авиабилета.

4. Если выбрано действие ВЫПОЛНИТЬ, выполняется подпоток S-4: реализовать заказ авиабилета.

5. Если выбрано действие ВЫХОД, элемент Use Case заканчивается.

#### Подпоток

**S-1: создать заказ авиабилета.** Система отображает диалоговое окно, содержащее поля для пункта назначения и даты полета. Покупатель вводит пункт назначения и дату полёта (E-2). Система отображает параметры авиарейсов (E-3). Покупатель выбирает авиарейс. Система связывает покупателя с выбранным авиарейсом (E-4). Возврат к началу элемента Use Case.

**S-2: удалить заказ авиабилета.** Система отображает параметры заказа. Покупатель подтверждает решение о ликвидации заказа (E-5). Система удаляет связь с покупателем (E-6). Возврат к началу элемента Use Case.

**S-3: проверить заказ авиабилета.** Система выводит (E-7) и отображает параметры заказа авиабилета: номер рейса, пункт назначения, дата, время, место, цену. Когда покупатель указывает, что он закончил проверку, выполняется возврат к началу элемента Use Case.

**S-4: реализовать заказ авиабилета.** Система запрашивает параметры кредитной карты покупателя. Покупатель вводит параметры своей кредитной карты (E-8). Возврат к началу элемента Use Case.

### **Альтернативные потоки**

**Е-1: введён неправильный ID-номер покупателя.** Покупатель может повторить ввод ID-номера или прекратить элемент Use Case.

**Е-2: введены неправильные пункт назначения/дата полёта.** Покупатель может повторить ввод пункта назначения/даты полёта или прекратить элемент Use Case.

**Е-3: нет подходящих авиарейсов.** Покупатель информируется, что в данное время такой полет невозможен. Возврат к началу элемента Use Case.

**Е-4: не может быть создана связь между покупателем и авиарейсом.** Информация сохраняется, система создаст эту связь позже. Элемент Use Case продолжается.

**Е-5: введён неправильный номер заказа.** Покупатель может повторить ввод правильного номера заказа или прекратить элемент Use Case.

**Е-6: не может быть удалена связь между покупателем и авиарейсом.** Информация сохраняется, система будет удалять эту связь позже. Элемент Use Case продолжается.

**Е-7: система не может вывести информацию заказа.** Возврат к началу элемента Use -Сазе.

**Е-8: некорректные параметры кредитной карты.** Покупатель может повторить ввод параметров карты или прекратить элемент Use Case.

Таким образом, в данной спецификации зафиксировано, что внутри элемента Use Case находится один основной поток и двенадцать вспомогательных потоков действий.

### **Комментарий к приведённому примеру**

В настоящем приложении приведён пример оформления текстового алгоритма реализации прецедента без использования таблиц, заимствованный из [6]. Заметим следующее: 1) условия выполнения вспомогательных потоков (подпотоков и альтернативных потоков) не помещены в квадратные скобки, что противоречит традиции (прагматике) применения UML-моделирования; 2) потоки ошибок названы альтернативными потоками, что хотя семантически не является некорректным, но также противоречит традиции; 3) места возникновения потенциальных ошибок помечены ссылками (в круглых скобках) на потоки шибок, а не оформляются как специальный пункт потока, что повышает компактность записи текста алгоритма.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Белов В.В., Чистякова В.И. Проектирование информационных систем: учебник – М. : КУРС, 2018. – 400 с. ISBN 978-5-906923-53-0 (КУРС) (45 экз. в БФ РГРТУ).

2. Белов В.В., Чистякова В.И. Проектирование информационных систем: учебник для студ. учреждений высш. образования / Под ред. В.В. Белова. – 2-е изд., стер. – М.: Издательский центр «Академия, 2015. – 352 с. (Сер. Бакалавриат). ISBN 978-5-4468-2440-3 (132 экз. в БФ РГРТУ)

3. Иванов, Денис Юрьевич. Унифицированный язык моделирования UML [Электронный ресурс]: учебное пособие для вузов по направлению подготовки "Системный анализ и управление" / Д.Ю. Иванов, Ф.А. Новиков; Санкт-Петербургский государственный политехн. ун-т. – Электрон. текстовые дан. (1 файл: 1,83 Мб). – Санкт-Петербург, 2011. – Загл. с титул. экрана. – Электронная версия печатной публикации. – Свободный доступ из сети Интернет (чтение, печать, копирование). – Текстовый документ. – Adobe Acrobat Reader 7.0. Доступно по URL:<http://elib.spbstu.ru/dl/2962.pdf>,  
<http://elib.spbstu.ru/dl/2962.pdf/download/2962.pdf>

4. Каюмова А.В. Визуальное моделирование систем в StarUML: Учебное пособие/ А.В. Каюмова. Казань. – Казанский федеральный университет, 2013. – 104 с.

5. OMG® Unified Modeling Language® (OMG UML® ). Version 2.5.1// OMG Document Number: formal/2017-12-05. Normative URL: <https://www.omg.org/spec/UML/> – 796 pp.

6. Орлов С.А. Программная инженерия. Технологии разработки программного обеспечения: Учебник для вузов. –5-е изд. обновл. и доп. Стандарт третьего поколения. – СПб.: Питер, 2016. – 640 с. Электрон. текстовые дан. (1 файл : 37,58 Мб). — Текстовый документ. — Adobe Acrobat Reader, Internet Explorer. Доступно по URL <https://www.twirpx.com/file/2378219/>.