



## РОЛЬ ПРАВИЛЬНОГО МОДЕЛИРОВАНИЯ ПРИ ПРОГРАММНОМ РЕШЕНИИ ЛОГИЧЕСКИХ ЗАДАЧ

**Жомуродов Дустмурод Мамасолиевич**

**Улашев Асроржон Насриддинович**

Джизакский филиал Национального университета

Узбекистана имени Мирзо Улугбека

[dustmurod@jbnuu.uz](mailto:dustmurod@jbnuu.uz)

**Аннотация:** Статья посвящена важности правильного моделирования при решении логических задач с помощью программных средств. Подчеркивается, что моделирование играет ключевую роль в формализации, структурировании и оптимизации задач, повышая эффективность и точность программных решений. Рассматриваются основные этапы моделирования: анализ проблемы, формализация задачи, определение переменных и отношений, создание абстракции, выбор метода решения и реализация модели. Приведены примеры успешного применения моделирования, такие как решение sudoku и планирование маршрутов. В заключении обсуждаются долгосрочные преимущества правильного моделирования, включая снижение рисков и повышение качества программного обеспечения.

**Ключевые слова:** Алгоритмы, качество программного обеспечения, логические задачи, моделирование, оптимизация, программное обеспечение, программные решения, формализация задачи, этапы моделирования, эффективность.

### Введение

Моделирование играет важнейшую роль в разработке программных решений для логических задач. В современном мире, где сложность задач растет с каждым днем, правильное моделирование позволяет существенно упростить процесс их





решения. Моделирование - это метод, который позволяет разработчикам формализовать и структурировать задачи, делая их решение более понятным и управляемым. В данной статье мы рассмотрим, почему моделирование является неотъемлемой частью разработки программного обеспечения, какие этапы включает в себя этот процесс, и приведем примеры успешного применения моделирования в различных областях.

Прежде чем приступить к программированию, важно понять и описать задачу, которая должна быть решена. Моделирование помогает разработчикам:

1. Формализовать задачу: Перевести реальную проблему на язык, понятный компьютеру.
2. Структурировать решение: Разделить сложную задачу на более мелкие и управляемые части.
3. Выбрать правильные методы: Определить, какие алгоритмы и структуры данных лучше всего подходят для решения данной задачи.
4. Избежать ошибок: Понять и учесть все возможные ограничения и условия задачи, что помогает избежать многих ошибок на этапе разработки.

Моделирование включает в себя несколько ключевых этапов, каждый из которых важен для успешного решения задачи. Давайте рассмотрим каждый из этих этапов более подробно.

## Понимание задачи

Перед тем как приступить к моделированию, важно глубоко понять задачу, которая стоит перед вами. Это ключевой этап, от которого зависит успех всего дальнейшего процесса моделирования и решения. В контексте математического моделирования понимание задачи включает в себя несколько важных шагов:

### 1. Анализ проблемы

Первым шагом является детальный анализ проблемы. Необходимо собрать всю возможную информацию о задаче и понять ее контекст. Важно выяснить следующие моменты:





**Определение проблемы:** четко сформулируйте, что именно нужно решить. Например, если задача связана с оптимизацией маршрутов доставки, определите, что именно оптимизируется (время, стоимость, расстояние).

**Цели и требования:** определите конечные цели задачи и требования, которым должно соответствовать решение. Это могут быть как количественные цели (например, минимизация затрат), так и качественные (например, улучшение удовлетворенности клиентов).

**Ограничения:** выявите все ограничения, которые необходимо учитывать. Например, ограничения на максимальное расстояние, грузоподъемность транспортных средств, время доставки и т.д.

## 2. Формулировка задачи

После анализа необходимо формализовать задачу. Формулировка задачи в математическом моделировании включает в себя следующие элементы:

**Определение переменных:** выделите все ключевые переменные задачи. Переменные могут быть как непрерывными (например, расстояние, время), так и дискретными (например, количество транспортных средств).

**Формулировка уравнений и неравенств:** определите математические уравнения и неравенства, которые описывают взаимосвязи между переменными. Например, уравнения могут описывать зависимость затрат от расстояния, а неравенства - ограничения на грузоподъемность.

**Целевая функция:** определите целевую функцию, которую нужно оптимизировать. Это может быть функция затрат, времени, прибыли и т.д. Например, в задаче оптимизации маршрутов целевой функцией может быть минимизация общего расстояния.

**Модель данных:** определите, какие данные необходимы для построения модели, и как они будут использоваться. Это могут быть данные о расстояниях между пунктами, стоимости перевозок, графике работы и т.д.

**Пример:** Оптимизация логистики





Рассмотрим пример задачи оптимизации логистики для доставки товаров. Пусть у нас есть несколько складов и множество пунктов доставки. Нам нужно оптимизировать маршруты доставки, чтобы минимизировать общие затраты на транспортировку.

### Анализ проблемы:

Определение проблемы: необходимо оптимизировать маршруты доставки товаров от складов к пунктам доставки.

Цели и требования: минимизировать общие затраты на транспортировку, учитывая ограничения по времени доставки и грузоподъемности транспортных средств.

Ограничения: Ограничения на максимальное расстояние маршрутов, максимальное время доставки, вместимость транспортных средств.

### 2. Формулировка задачи:

Определение переменных: пусть  $x_{ij}$  – бинарная переменная, которая равна 1, если маршрут идет от склада  $i$  к пункту доставки  $j$ , и 0 в противном случае. Пусть  $d_{ij}$  – расстояние между складом  $i$  и пунктом доставки  $j$ .

Формулировка уравнений и неравенств:

- Ограничение на использование транспортных средств:

$$\sum_j x_{ij} \leq C_i, \quad \forall i$$

где  $C_i$  – вместимость транспортного средства  $i$ .

- Ограничение на время доставки:

$$\sum_i x_{ij} t_{ij} \leq T_j, \quad \forall j$$

где  $t_{ij}$  – время доставки от склада  $i$  к пункту доставки  $j$ , а  $T_j$  – максимальное допустимое время доставки к пункту  $j$ .

Целевая функция:





$$\min \sum_i \sum_j c_{ij} x_{ij}$$

где  $c_{ij}$  – стоимость доставки от склада  $i$  к пункту доставки  $j$ .

Модель данных: необходимо собрать данные о расстояниях  $d_{ij}$ , времени доставки  $t_{ij}$ , стоимости  $c_{ij}$ , вместимости транспортных средств  $C_i$  и допустимом времени доставки  $T_i$ .

Таким образом, формализация задачи позволяет четко определить, что нужно решить и какими методами это можно сделать. Это критически важный этап, от которого зависит успешное построение и реализация математической модели.

## Этапы моделирования

Моделирование логических задач, особенно с использованием математических методов, включает несколько ключевых этапов. Каждый из них важен для создания точной и эффективной модели, которая поможет найти решение задачи. Рассмотрим эти этапы подробнее:

### 1. Определение переменных и отношений

Первый шаг в математическом моделировании – это определение всех переменных, которые описывают задачу, и установление отношений между ними. Это включает:

- **Идентификация переменных:** Переменные могут представлять различные аспекты задачи. Например, если мы моделируем логистическую задачу, переменные могут представлять количество товаров, время доставки, расстояние и т.д.
- **Определение отношений:** Эти отношения описывают, как переменные взаимодействуют друг с другом. В математическом моделировании это часто выражается через уравнения или неравенства. Например, если  $x$  и  $y$  – это количества товаров, то  $x + y$  может быть ограничено вместимостью склада.

### 2. Создание абстракции

Создание абстракции – это процесс упрощения реальной задачи, сохраняя её основные характеристики. Это делается для того, чтобы задача стала управляемой и решаемой с помощью математических методов. Абстрагирование включает:





- Удаление несущественных деталей: это позволяет сконцентрироваться на ключевых аспектах задачи. Например, при моделировании потока транспорта можно не учитывать мелкие особенности дорог, а сосредоточиться на основных маршрутах.
- Формализация задачи: Преобразование задачи в форму, удобную для анализа и решения. Это может включать представление задачи в виде систем уравнений, графов, матриц и т.д.

### 3. Выбор метода решения

После создания абстрактной модели задачи необходимо выбрать подходящий метод её решения. Выбор метода зависит от типа задачи и её характеристик. В математическом моделировании часто используются следующие методы:

- Алгоритмы оптимизации: например, линейное программирование, динамическое программирование, методы ветвей и границ.
- Численные методы: например, методы итерации для нахождения решений систем уравнений.
- Алгоритмы поиска: например, алгоритмы поиска в графах, такие как алгоритм Дейкстры для нахождения кратчайшего пути.
- Методы теории вероятностей и статистики: используются для задач, связанных с неопределенностью и случайными процессами.

### 4. Реализация

После выбора метода решения необходимо реализовать модель в программном коде. Этот этап включает:

- Программирование алгоритма: Написание кода, который реализует выбранный метод решения. Важно учитывать особенности задачи и обеспечить корректность и эффективность реализации.
- Валидация модели: Проверка модели на корректность. Это может включать тестирование на наборе тестовых данных, чтобы убедиться, что модель правильно решает задачу.
- Оптимизация кода: Улучшение производительности программы, чтобы она работала быстро и эффективно даже на больших наборах данных.

**Пример:** Моделирование задачи о наилучшем распределении ресурсов





Рассмотрим пример задачи о наилучшем распределении ресурсов, которую можно решить с помощью линейного программирования:

1. Определение переменных: пусть  $x_1, x_2, \dots, x_n$  – это количество различных ресурсов, которые нужно распределить.
2. Определение отношений: пусть  $c_1, c_2, \dots, c_n$  – это стоимость каждого ресурса, а  $a_{ij}$  – количество ресурса  $j$ , необходимое для выполнения работы  $i$ . Ограничения могут быть заданы в виде системы линейных уравнений или неравенств.
3. Создание абстракции: формулируем задачу в виде системы линейных уравнений и неравенств, описывающих ограничения, и функции цели  $z = \sum_{j=1}^n c_j x_j$ , которую нужно минимизировать или максимизировать.
4. Выбор метода решения: используем метод симплекс для решения задачи линейного программирования.
5. Реализация: Написание кода на языке программирования, например, Python, с использованием библиотеки для линейного программирования, такой как PuLP или SciPy.

Этапы моделирования – это последовательный процесс, который начинается с глубокого понимания задачи и заканчивается реализацией модели в программном коде. Правильное математическое моделирование позволяет создать точную и эффективную модель, которая поможет найти оптимальное решение сложной логической задачи.

## Примеры успешного моделирования

Моделирование нашло широкое применение в самых разных областях, от игр до сложных научных исследований. Рассмотрим три ярких примера успешного моделирования: решение sudoku, планирование маршрутов и автоматическое доказательство теорем.

### 1. Решение sudoku

Sudoku – это популярная логическая игра, которая представляет собой сетку 9x9, разделенную на меньшие квадраты 3x3. Цель игры состоит в том, чтобы





заполнить сетку цифрами от 1 до 9 таким образом, чтобы в каждой строке, столбце и квадрате 3x3 каждая цифра встречалась только один раз.

Моделирование задачи sudoku:

- Переменные: каждая ячейка сетки sudoku представляется переменной, которая может принимать значения от 1 до 9.
- Ограничения: для каждой строки, столбца и квадрата 3x3 создаются ограничения, чтобы каждое число от 1 до 9 появлялось только один раз.

Методы решения:

- Задача sudoku может быть формализована как задача удовлетворения ограничений (CSP - Constraint Satisfaction Problem). Алгоритмы для решения CSP, такие как метод возврата (backtracking) и улучшенные методы, использующие эвристики и пропаганду ограничений, эффективно решают задачи sudoku.

Преимущества:

- Формализация задачи в виде CSP позволяет использовать мощные алгоритмы, что делает решение быстрым и надежным.
- Модель легко адаптируется к различным уровням сложности и форматам sudoku.

## 2. Планирование маршрута

Планирование маршрута является важной задачей в логистике, транспорте и навигации. Цель задачи – найти наиболее оптимальный путь из одной точки в другую с учетом различных факторов, таких как расстояние, время и стоимость.

Моделирование задачи планирования маршрута:

- Переменные: Вершины графа представляют места (например, города, перекрестки), а ребра – дороги или маршруты между ними.
- Ограничения: Каждое ребро может иметь вес (например, расстояние или время), что влияет на выбор оптимального маршрута.

Методы решения:

- Алгоритмы поиска пути в графах, такие как алгоритм Дейкстры и алгоритм A\*, широко используются для нахождения кратчайших путей в взвешенных графах. Алгоритм Дейкстры особенно эффективен для графов с неотрицательными весами, в то время как A\* использует эвристики для ускорения поиска.





Преимущества:

- Эти алгоритмы обеспечивают нахождение оптимальных маршрутов с учетом заданных критериев.
- Модель может легко адаптироваться для учета дополнительных факторов, таких как трафик, дорожные условия и предпочтения пользователей.

### 3. Автоматическое доказательство теорем

Автоматическое доказательство теорем – это область искусственного интеллекта и математической логики, целью которой является создание программ, способных автоматически доказывать математические теоремы.

Моделирование задачи доказательства теорем:

- Переменные и формулы: Математические утверждения и теоремы представляются в виде логических формул, использующих предикаты, функции и константы.
- Ограничения: Законы логики и аксиомы служат ограничениями, которые должны быть выполнены для доказательства теоремы.

Методы решения:

- Логические исчисления, такие как исчисление высказываний и предикатов, используются для моделирования доказательств. Методы автоматического доказательства включают резолюцию, метод прямого доказательства, метод индукции и другие.
- Современные системы автоматического доказательства, такие как Coq, Isabelle и Z3, используют комбинации этих методов для поиска доказательств.

Преимущества:

- Автоматическое доказательство теорем позволяет находить доказательства сложных математических утверждений, которые могут быть трудны для человека.
- Эти системы могут проверять доказательства на корректность, что снижает вероятность ошибок в математических исследованиях.

Эти примеры демонстрируют, как правильное моделирование задач помогает находить эффективные решения в различных областях. Моделирование позволяет формализовать задачи, структурировать их и выбирать оптимальные методы для их решения, что делает процесс разработки программного обеспечения более управляемым и успешным.





## Преимущества правильного моделирования

- Повышение эффективности: хорошо спроектированные модели позволяют использовать эффективные алгоритмы, что сокращает время решения задачи.
- Улучшение понимания задачи: Моделирование помогает структурировать задачу и выявить скрытые взаимосвязи и зависимости.
- Легкость модификации: Модели проще адаптировать к изменяющимся требованиям и условиям, что повышает гибкость программного решения.

## Заключение

Правильное моделирование – это не просто этап в разработке программного обеспечения, а ключевой компонент, от которого зависит успех всего проекта. В заключении мы подведем итоги и обсудим основные причины, по которым моделирование является столь важным для решения логических задач.

## Основные преимущества правильного моделирования

### 1. Повышение эффективности:

- Оптимизация ресурсов: хорошо спроектированные модели позволяют использовать оптимальные алгоритмы и структуры данных, что значительно сокращает время выполнения программы и снижает потребление ресурсов.
- Уменьшение времени разработки: Ясная и четкая модель задачи помогает разработчикам быстрее и точнее реализовать решения, что сокращает время, необходимое на написание и тестирование кода.

### 2. Улучшение понимания задачи:

- Декомпозиция задачи: моделирование помогает разбить сложную задачу на более мелкие и управляемые части. Это упрощает анализ и решение задачи, позволяя сосредоточиться на каждом компоненте отдельно.
- Выявление скрытых взаимосвязей: при моделировании разработчики могут обнаружить скрытые взаимосвязи и зависимости между элементами задачи, что помогает более полно понять её структуру и поведение.

### 3. Легкость модификации:





- Адаптивность к изменениям: Модели проще адаптировать к изменяющимся требованиям и условиям. Это особенно важно в условиях, когда задачи и требования могут меняться в процессе разработки.
- Упрощение отладки и улучшений: Модели предоставляют четкую структуру, что облегчает процесс отладки и внесения улучшений в программный код.

#### Долгосрочные преимущества

##### 1. Снижение рисков:

- Минимизация ошибок: Моделирование помогает выявить и учесть все возможные ограничения и условия задачи на ранних этапах, что снижает вероятность возникновения ошибок в коде.
- Предсказуемость результатов: Модели позволяют предсказать поведение системы в различных ситуациях, что делает конечный продукт более надежным и предсказуемым.

##### 2. Повышение качества программного обеспечения:

- Модульность и повторное использование: Хорошо структурированные модели способствуют созданию модульного кода, который можно повторно использовать в других проектах, что повышает общую эффективность разработки.
- Документирование процесса: Модели служат отличной документацией, которая может быть использована для обучения новых членов команды и облегчения понимания системы в будущем.

#### Заключительные мысли

Правильное моделирование – это фундаментальный этап в решении логических задач с помощью программных средств. Оно позволяет разработчикам не только четко формулировать задачи, но и выбирать оптимальные методы их решения, создавая при этом эффективные и надежные программные продукты. Вложение времени и усилий в качественное моделирование на начальном этапе разработки окупается за счет повышения точности, эффективности и адаптивности конечного решения.

В условиях быстро меняющегося технологического ландшафта, правильное моделирование становится особенно важным. Оно помогает справляться с увеличивающейся сложностью задач, обеспечивая разработчиков необходимыми инструментами и методологиями для создания высококачественного программного





обеспечения. В конечном итоге, правильное моделирование – это инвестиция в будущее, которая обеспечивает устойчивый успех проектов и высокое качество конечного продукта.

### Список литературы:

1. Жомуродов Д., Мелиева М. ИННОВАЦИИ В ОБУЧЕНИИ: НОВЫЕ ПОДХОДЫ К РАБОТЕ С МАССИВАМИ В ОБРАЗОВАТЕЛЬНЫХ ПРОГРАММАХ //Uz-Conferences. – 2023. – Т. 1. – №. 1. – С. 867-872.
2. Jomurodov D., Meliyeva M. THE ADVANTAGES OF IMPLEMENTING AUTOMATED SYSTEMS IN COMPUTER SCIENCE LEARNING //International Scientific and Practical Conference on Algorithms and Current Problems of Programming. – 2023.
3. Zhomurodov D., Ulashev A., Tozhiyev A. THE SYSTEM FOR DETERMINING THE QUALIFICATIONS OF INDUSTRY EXPERTS //Евразийский журнал академических исследований. – 2023. – Т. 3. – №. 4 Special Issue. – С. 280-289.
4. Жомуродов Д. Интерактивное обучение кибербезопасности //Новый Узбекистан: наука, образование и инновации. – 2024. – Т. 1. – №. 1. – С. 149-155.
5. Жомуродов Д. Оптимизация и улучшение производительности алгоритма шифрования Blowfish //Новый Узбекистан: наука, образование и инновации. – 2024. – Т. 1. – №. 1. – С. 155-158.
6. Жомуродов Д., Улашев А. Дополненная реальность в робототехнике: объединение миров ради будущего //Информатика и инженерные технологии. – 2023. – Т. 1. – №. 1. – С. 49-52.
7. Kayumov O. et al. ELECTRONIC PLATFORM FOR RECOGNITION AND TEACHING OF SIGN LANGUAGE PICTURES BASED ON UZBEK GRAMMAR //International Journal of Contemporary Scientific and Technical Research. – 2023. – С. 263-268.
8. Dustmurod J., Rayxona A. UTILIZING VR AND AR TECHNOLOGIES FOR CREATING ELECTRONIC LEARNING RESOURCES //Uz-Conferences. – 2023. – Т. 1. – №. 1. – С. 861-866.