|  |  |
| --- | --- |
| Министерство науки и высшего образования Российской Федерации  ОРСКИЙ ГУМАНИТАРНО-ТЕХНОЛОГИЧЕСКИЙ ИНСТИТУТ (ФИЛИАЛ)  ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ  «ОРЕНБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  Факультет среднего профессионального образования | |
| **Курсовая работа**  по междисциплинарному курсу «Технология разработки программного обеспечения»  профессионального модуля «Осуществление интеграции программных модулей»    **Разработка программного обеспечения для автоматизации расчётов в классической механике**  Пояснительная записка  ОГУ 09.02.07. 3024.157 ПЗ | |
|  | Руководитель работы  преподаватель высшей категории  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ж. В. Михайличенко  «\_\_\_» \_\_\_\_\_­\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.  Студент группы 22ИСП-2  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ A. A. Тажбаев  «\_\_\_» \_\_\_\_\_­\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г. |
| Орск 2024 | |

|  |  |
| --- | --- |
| Утверждаю  председатель ПЦК дисциплин профессионального цикла | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись | Ж.В. Михайличенко |
| «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г. | |

**ЗАДАНИЕ**

**на выполнение курсовой работы**

студенту \_\_\_\_\_Тажбаеву Амиру Амангельдыевичу\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

фамилия, имя, отчество

по специальности \_\_\_09.02.07 Информационные системы и программирование\_\_\_\_\_\_\_\_\_\_\_\_\_\_

по междисциплинарному курсу \_\_Технология разработки программного обеспечения\_\_\_\_\_\_\_\_\_\_

Тема работы: \_Разработка программного обеспечения для автоматизации расчётов в классической механике\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Срок сдачи студентом работы «10» \_\_июня\_\_\_\_ 2024 г.
2. Цель и задачи работы \_\_Разработать программный продукт, позволяющий решать 7-10 задач из раздела физики «Классическая механика» с использованием различных входных данных\_\_\_\_
3. Исходные данные к работе: \_\_Учебники и интернет-источники по технологии разработки программного обеспечения и практикумы по физике\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
4. Перечень вопросов, подлежащих разработке: \_ а) изучить предметную область, выполнить анализ требований к программному обеспечению, составить техническое задание на разработку; б) выполнить проектирование системы с помощью CASE-средств; в) для решения поставленной задачи реализовать оконное приложение на языке C# и протестировать его; г) сформулировать предложения по внедрению, эксплуатации и сопровождению разработанного программного обеспечения. Сделать выводы по результатам проделанной работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
5. Перечень графического (иллюстративного) материала: таблицы, графики, рисунки, схемы, отражающие теоретический материал и программную реализацию поставленной задачи\_\_\_\_\_\_\_

Дата выдачи и получения задания

Руководитель «19» \_февраля\_\_\_\_\_ 2024 г. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_Ж.В. Михайличенко\_\_\_

подпись инициалы, фамилия

Студент «19» \_февраля\_\_\_\_\_ 2024 г. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_А.А. Тажбаев\_\_\_\_\_\_\_\_

подпись инициалы, фамилия

**Аннотация**

Изм.

Лист

№ докум.

Подпись

Дата

Лист

3

ОГУ 09.02.07. 3024. 157 ПЗ

Разраб.

Тажбаев A.A

Провер.

Михайличенко Ж

Реценз.

Н. Контр.

Утверд.

Разработка программного обеспечения для автоматизации расчётов в классической механике

Лит.

Листов

\*

22ИСП-2

В курсовой работе по междисциплинарному курсу «Технология разработки программного обеспечения» профессионального модуля «Осуществление интеграции программных модулей» проведена разработка \*\*\*\*\*.

В первой главе курсовой работы \*\*\*

Во второй главе курсовой работе \*\*\*

В третьей главе курсовой работы \*\*\*

Пояснительная записка содержит \*\* страницы, в том числе \*\* рисунков, \*\* таблиц, \*\* источников, 1 приложение.

Разработка приложения выполнена \*\*\*.

**Содержание**

[Введение 7](#_Toc168392682)

[1 Анализ требований и проектирование программного продукта 8](#_Toc168392683)

[1.1 Анализ предметной области 8](#_Toc168392684)

[1.2 Техническое задание на разработку 9](#_Toc168392685)

[1.3 Проектирование 12](#_Toc168392686)

[2 Разработка и тестирование программного продукта 16](#_Toc168392687)

[2.1 Обоснование программы средств реализации 16](#_Toc168392688)

[2.2 Разработка пользовательского интерфейса 17](#_Toc168392689)

[2.3 Алгоритмизация и программирование 24](#_Toc168392690)

[2.5 Тестирование 26](#_Toc168392691)

[3 Рекомендации по внедрению, эксплуатации и сопровождению программного продукта 27](#_Toc168392692)

[3.1 Руководство пользователя 27](#_Toc168392693)

[3.2 Планы внедрения и сопровождения 32](#_Toc168392694)

[Заключение 33](#_Toc168392695)

[Список использованных источников 34](#_Toc168392696)

[Приложение А 35](#_Toc168392697)

# Введение

Целью данной курсовой работы заключается в разработке программного продукта, способного автоматизировать процесс расчетов в классической механике, упростить выполнение сложных математических операций и повысить точность результатов. Ещё совсем недавно основными инструментами инженера были калькулятор и чертёжная доска. Расчёты занимали немало рабочего времени. Например, большинство инженерных расчётов проводятся в нескольких приближениях, то есть один и тот же алгоритм вычислений повторяется несколько раз, но каждый раз с новыми, уточнёнными данными. Инженер вынужден был повторять на калькуляторе вычислительные операции каждого приближения снова и снова. Выполнение чертежей также было нелёгким занятием. Нередки были случаи, когда при возникновении ошибки приходилось перечерчивать заново весь чертёж.

К сожалению, на некоторых предприятиях, работающих «по старинке», такая ситуация существует до сих пор. Однако, в современных условиях рыночной экономики, такой подход, несомненно, не позволит предприятию быть конкурентоспособным и получать прибыль. При размещении заказа на выполнение проектных работ одним из основных критериев является минимальное время выполнения проекта.

Автоматизация инженерной работы (расчётов, выполнения чертежей и текстовых документов) позволяет сократить время выполнения проекта в несколько раз. Для этого требуется оборудовать место работы инженера персональным компьютером и установить на него соответствующее программное обеспечение. Возможности компьютера позволяют использовать его как средство автоматизации инженерной и научной работы.

Для решения сложных расчётных задач используют программы, написанные специально. В то же время, в инженерной и научной работе встречается широкий спектр задач ограниченной сложности, для решения которых можно использовать универсальные средства. К такого рода задачам относятся:

- подготовка научно-технических документов, содержащих текст и формулы, записанные в привычной для специалистов форме;

- вычисление результатов математических операций, в которых участвуют числовые константы, переменные и размерные физические величины;

- статистические расчёты и анализ данных;

- построение двумерных и трёхмерных графиков;

- дифференцирование и интегрирование, аналитическое и численное;

- решение дифференциальных уравнений;

- проведение серий расчётов с разными значениями начальных условий и других параметров;

- выполнение чертежей.

К универсальным программам, пригодным для решения таких задач, относятся программы MS Word, MathCad, MS Excel.

# Анализ требований и проектирование программного продукта

## Анализ предметной области

Автоматизация расчётов – это когда пользователь вводит начальные данные, а некоторая [программа](https://www.usu.kz/wiki/programma.php) сама проводит расчёты на основе заложенных в неё [алгоритмов](https://www.usu.kz/wiki/algoritm.php) и выдаёт конечный результат пользователю. Автоматизация расчётов помогает избавится от ошибок человека, которые могут быть допущены при расчёте, особенно когда это большая и сложная формула. Автоматизация расчётов применяется в автоматических системах, которые разрабатываются для [автоматизации рабочего места](https://www.usu.kz/wiki/avtomatizaciya_rabochego_mesta.php). Автоматические системы повышают эффективность работы сотрудников, избавляя его от однообразной работы

Разработка программного обеспечения для автоматизации расчётов в классической механике включает применение законов классической механики для решения задач движения тел и взаимодействия между ними. Классическая механика описывает движение объектов под воздействием сил и использует математические модели и законы, такие как уравнения Ньютона, законы сохранения энергии, импульса, инерции и так далее.

Автоматизация расчётов в классической механике с помощью программного обеспечения позволяет эффективно выполнять сложные математические операции и обработку данных. Автоматизация этих процессов позволяет минимизировать ошибки вычислений, тем самым позволяя создавать более реалистичные модели поведения тел.

Это программное обеспечение поможет для работы инженеров, учёных и студентов, помогая им в решении задач движения тел и моделирования физических процессов.

Разработка подобного программного обеспечения имеет большое значение для оптимизации работы специалистов в области физики и математики, облегчая процессы моделирования и анализа работы физических систем.

Целью может быть разработка программного обеспечения, которое позволит автоматизировать расчёты в классической механике, повысить их точность и сократить время на выполнение вычислений. Задачи могут включать разработку алгоритмов решения задач классической механики, создание пользовательского интерфейса для ввода данных и отображения результатов расчетов, тестирование и отладка программы, анализ возможных ошибок и путей их устранения. Также важно рассмотреть существующие подходы к решению задач классической механики и определить, какие из них наиболее эффективны и применимы для автоматизации.

Для достижения цели курсовой работы выбраны следующие задачи из раздела классической механики для программной реализации:

1 Задача о колебаниях: найти период колебаний математического маятника, если известны его длина и ускорение свободного падения.

2 Задача о законах Ньютона: определить силу, действующую на тело массой m, если известны его ускорение и масса.

3 Автомобиль массой m движется со скоростью v. Найдите кинетическую энергию автомобиля.

4 Задача о кинетической энергии: найти кинетическую энергию тела массой m, движущегося со скоростью v.

5 Задача о потенциальной энергии: вычислить потенциальную энергию системы, состоящей из нескольких тел, находящихся на различной высоте.

6 Тело массой m, которое движется по горизонтальной поверхности. На это тело действует ускорение a, и дополнительная сила трения F. Найдите результирующую силу, действующую на тело.

7 Задача о движении тела по прямой: определить ускорение тела, если известны начальная скорость, время движения и пройденное расстояние.

## Техническое задание на разработку

Техническое задание на разработку программного обеспечения – это документ, который содержит подробное описание требований к программному продукту. Техническое задание на разработку автоматизации расчётов в классической механике составлено согласно ГОСТ 34.602-2020 «Информационные технологии. Комплекс стандартов на автоматизированной системы. Техническое задание на создание автоматизированной системы» и представлено ниже.

1 Общие сведения.

а) Полное наименование автоматизированной системы (АС): «Автоматизированная система расчётов в классической механике»;

б) Наименование заказчика: факультет среднего профессионального образования Орского гуманитарно-технологического института (филиала ОГУ) в лице преподавателя высшей категории Михайличенко Ж.В.;

в) Наименование разработчика: студент второго 2 курса группы 22ИСП-2 Тажбаев A.A.;

г) Документ, на основании которого создается АС: протокол закрепления тем курсовых работ по дисциплине «Технология разработки программного обеспечения» от 19.02.2024 года;

д) Дата начала работ: 19.02.2024 год;

е) Дата окончания работ: 10.06.2024 год.

2 Цели и назначения создания автоматизированной системы.

а) Цели создания АС: упрощение и ускорение расчётов, повышение точности, улучшение доступности, решение сложных задач, повышение эффективности исследований.

б) Назначение АС: автоматизация выполнения расчётов, анализ данных, моделирование процессов, обработка результатов в классической механике. Предназначена для помощи исследователям, инженерам, учёным, студентам.

3 Характеристика объекта автоматизации.

Это программное средство позволяет проводить сложные математические расчёты, моделировать различные физические системы и анализировать их поведение, а также облегчает работу учёных, инженеров и студентов, занимающихся изучением физических явлений.

Использование такого программного обеспечения позволяет ускорить процесс выполнения расчётов, снизить вероятность ошибок и повысить точность получаемых результатов.

4 Требования к автоматизированной системе:

а) Требования к АС в целом: надёжность и стабильность работы, высокая производительность для быстрого выполнения расчетов, простота использования и интуитивно понятный интерфейс, возможность автоматизации процессов расчета, гибкость и возможность настройки под различные задачи.

б) Требования к функциям, выполняемым АС:

* выбор задачи для расчётов;
* ввод исходных данных;
* проведение расчётов по заданной формуле;
* вывод результатов на экран монитора;
* защита данных от неверного ввода.

в) Требование к видам обеспечений АС.

* Требования к математическому обеспечению: метод классической механики используется для моделирования движения тел в системе в течение времени. Алгоритмы обработки данных используются для анализа результатов расчетов.
* Требования к информационному обеспечению: хранение и обработка данных о телах.
* Требования к лингвистическому обеспечению: язык интерфейса системы и диалогов - русский.
* Требования к программному обеспечению: для обеспечения работы в области классической механики требуется использовать определенное программное обеспечение. Используемое программное обеспечение включает в себя операционную систему Windows, средства моделирования бизнес-процессов «Ramus Educational», язык программирования С#, среду программирования Visual Studio и средства документирования Microsoft Word для создания руководства пользователя Dr. Explain.
* Требования к техническому обеспечению: для полноценного функционирования автоматизированной системы в области классической механики необходимо, современный, средне персональный компьютер с установленной операционной системой (Windows 11, Windows 10).
* Требования к организационному обеспечению: пользователь будет взаимодействовать с системой, он сможет вводить исходные данные с клавиатуры и получать результаты расчётов на экран монитора;

г) Общие технические требования к АС.

* Требования к численности и квалификации пользователей: система должна предназначена только для одного пользователя, имеющего базовую грамотность и значение формул и законов классической физики.
* Требования к защите данных: защита от неправильного ввода символов.

5 Состав и содержание работ по созданию автоматизированной системы.

В таблице 1 представлены этапы разработки АС расчётов в классической механике.

Таблица 1 – Этапы разработки АС

|  |  |  |
| --- | --- | --- |
| Номер и название этапа разработки | Сроки выполнения | Содержание |
| 1 Анализ требований | 19.02.2024 – 10.03.2024 | Анализ предметной области, изучение программных аналогов, разработка технического задания на создания АС |
| 2 Проектирование | 15.03.2024 –25.03.2024 | Создание интерфейсов пользователя и алгоритмов расчётов, разработка архитектуры системы |
| 3Программирование | 10.04.2024 –  20.04.2024 | Создание кода с учётом разработанной архитектуры и требований |
| 4 Тестирование | 30.04.2024 –31.04.2024 | Выявление ошибок, проверка точности расчётов |
| 5 Эксплуатация | 12.05.2024 –24.05.2024 | Обучение новый пользователей, решение возникающих проблем |
| 6 Внедрение и сопровождения | 5.06.2024 – 10.06.2024 | Внедрение системы в окружение, мониторинг работы системы, дальнейшая развитие и сопровождение |

6 Порядок разработки автоматизированной системы: определение требований к системе на основе анализа задач классической механики, которые требуется автоматизировать. Проектирование структуры программного обеспечения, включая выбор языка программирования, архитектуры системы и базы данных. Разработка и тестирование программного обеспечения для расчетов в классической механике. Внедрение системы и настройка ее под конкретные потребности пользователя. Обучение пользователей работе с системой и предоставление технической поддержки. Поддержка и сопровождение системы, включая регулярные обновления и исправление ошибок.

7 Порядок контроля АС: Проведение тестирования программного обеспечения на всех этапах разработки. Аудит безопасности системы для защиты данных и предотвращения несанкционированного доступа. Мониторинг работы системы для выявления проблем и оптимизации производительности. Регулярное обновление и поддержка системы для обеспечения ее надежной работы.

8 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу автоматизированной системы в действие. Для использования автоматизированной системы требуется установка на персональный компьютер пользователя.

9 Требования к документированию. Разработка технической документации по системе, включая описание функционала, инструкции по использованию и администрированию. Создание руководства пользователя с пошаговыми инструкциями по работе с системой. Ведение журналов изменений и ошибок для последующего анализа и улучшения работы системы. Документирование процессов работы системы для обеспечения прозрачности и эффективности использования.

10 Источники разработки.

* Протокол закрепления тем курсовых работ по дисциплине «Технология разработки программного обеспечения»;
* ГОСТ 34.602-2020 «Информационные технологии. Комплекс стандартов на автоматизированной системы. Техническое задание на создание автоматизированной системы»;
* Работы студенческие. Общие требования и правила оформления. СТО 02069024.101 – 2015. – Оренбург: Изд-во ОГУ, 2015. – 89 с.
* Добавить ГОСТ на блок-схемы
* Добавить ГОСТ на этапы разработки АС

## Проектирование

Функциональное проектирование автоматизированной системы расчёта в классической механике позволяет значительно ускорить процесс выполнения сложных математических операций и анализа физических явлений. Это способствует более точным и быстрым результатам, что важно для научных и инженерных исследований, а также для разработки новых технологий и материалов.

Функциональная модель на основе метода IDEF0 представляет собой абстрактное описание функций и процессов в системе. Этот метод позволяет структурировать и описать основные функции, их взаимосвязи и зависимости. Построение функциональной модели с помощью IDEF0 упрощает коммуникацию между участниками проекта и позволяет выявить узкие места в системе для оптимизации процессов. Анализ функциональной модели с использованием IDEF0 помогает лучше понять взаимодействие системы с окружающими системами. Использование стандартизированных методов IDEF0 способствует более эффективному проектированию, оптимизации и моделированию системы.

Ramus — это программа для построения визуальных диаграмм, используемых для наглядного отображения различных бизнес процессов.

Контекстная модель автоматизированной системы расчёта в классической механике, которая показана ниже представляет собой структурированное описание того, как система взаимодействует с внешним миром. В качестве входных компонентов выступает «Исходные данные». Управление осуществляется с помощью «законов физики», которые определяют правила и принципы, по которым система должна выполнить задачу.  Выходными компонентами являются «решённые задачи», которые представляют собой результат работы системы по обработке и анализу входных данных с учётом законов физики.  Механизм выполнения задачи осуществляется с помощью «персонального компьютера (ПК)», который выполняет вычисления и обработку данных в соответствии с заданными законами физики. «Пользователь» является человеком, который взаимодействует с системой, предоставляя и получая информацию, необходимую для выполнения задачи.

Родительская диаграмма IDEF0 показана на рисунке 1.

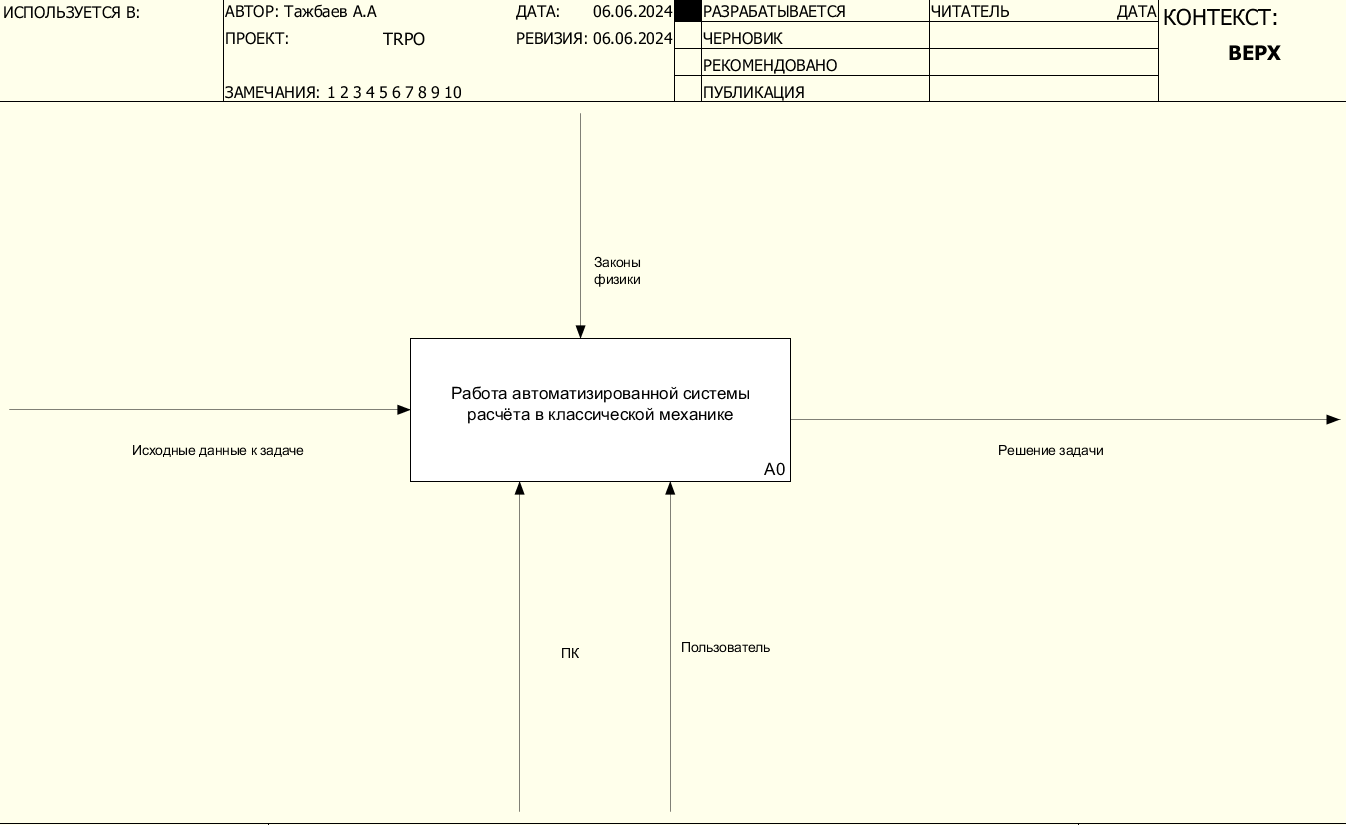


Рисунок 1 – Родительская диаграмма IDEF0

В данном блоке А0 «Работа автоматизированной системы расчётов в классической механике» входят такие компоненты как: исходные данные к задаче, механизм: ПК, пользователь, управление: законы физики, выход: решённые задачи.

 Для более глубокого понимания функциональной модели, необходимо декомпозировать блок А0 в диаграмму первого уровня, включающую следующие компоненты:

Блок А1 «Выбор задачи»: входным компонентом является список номеров задач, механизм осуществляется пользователем, который выбирает номер задачи из предложенного списка, взаимодействуя с компьютером; на выходе получаем номер задачи;

   Блок А2 «Ввод исходных данных к задаче»: входными компонентами являются условия задачи и номер выбранной задачи; механизм: пользователь, который вводит необходимые данные для решения задачи, указывая условия задачи и известные значения с помощью компьютера; на выходе получаем известные значения;

    Блок А3 «Выполнение расчётов по формулам»: входные компоненты использованы в виде формул и известных значений; управление: программа на компьютере, которая выполняет расчёты по соответствующим формулам, используя законы физики для выполнения математических операций; механизм: ПК; на выходе получаем результаты расчётов;

    Блок А4 «Вывод результатов расчёта»: входные компоненты использованы в виде результатов расчётов; в качестве механизма используется компьютер, который выводит полученные результаты на экран для ознакомления пользователя, позволяя ему оценить правильность выполненных расчётов и получить решённую задачу. Диаграмма декомпозиции первого уровня показана на рисунке 2.

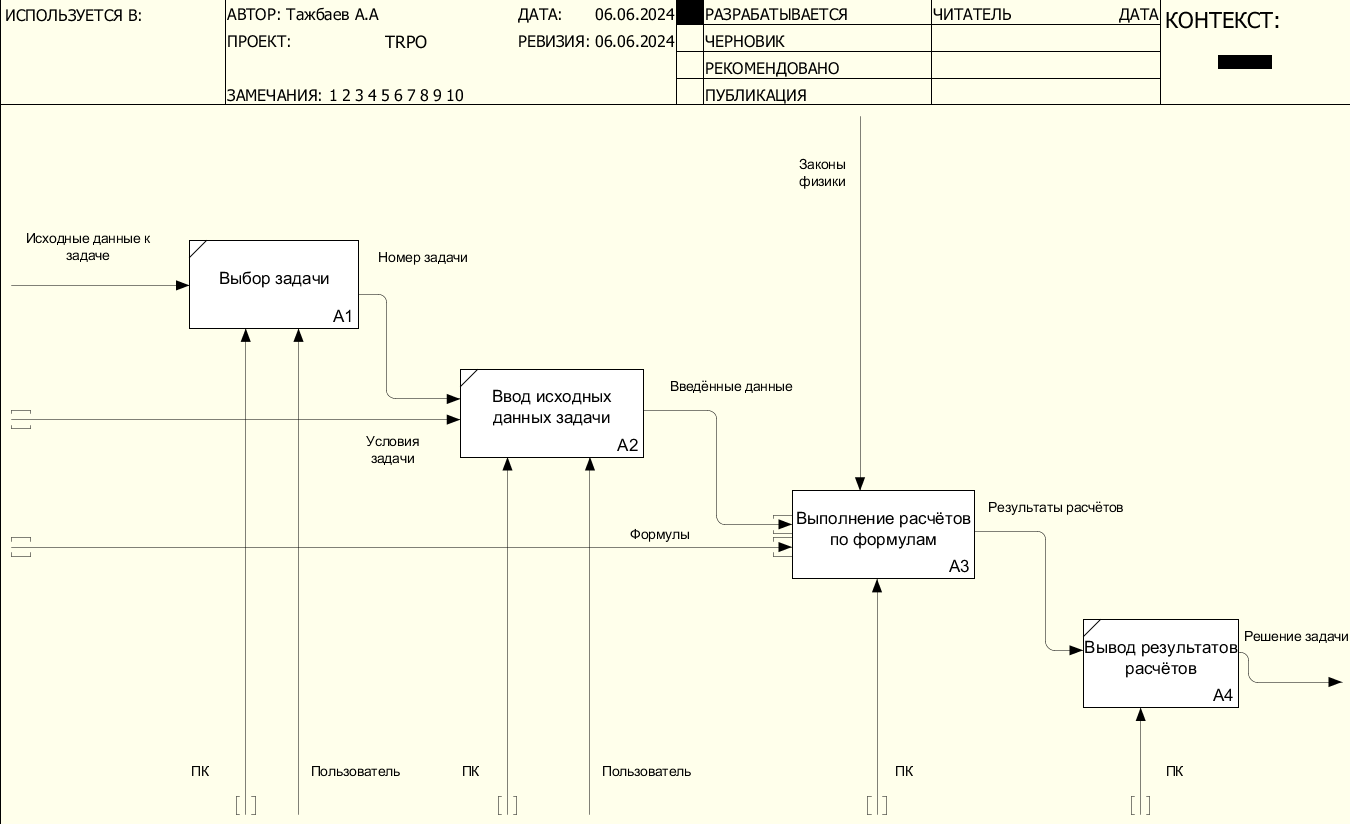


Рисунок 2 – Диаграмма декомпозиции IDEF0 первого уровня

Проведём декомпозицию блока А3 «Выполнение расчётов по формуле» на функциональные блоки А31, А32, А33, А34.

А31 – Здесь происходит получение второго закона Ньютона для расчёта силы действующую на массу.

* Вход: второй закон Ньютона;
* Выход: необходимые переменные;
* Механизм: персональный компьютер.

A32 – Здесь подставляют значения.

* Вход: необходимые переменные, масса тела(m), ускорение(a);
* Выход: готовое выражение;
* Механизм: персональный компьютер.

А33 – Здесь выполняется расчёт силы действующую на массу.

* Вход: готовое выражение;
* Выход: результаты расчётов;
* Механизм: персональный компьютер;
* Управление: законы физики, второй закон Ньютона.

Диаграмма декомпозиции второго уровня показана на рисунке 3.

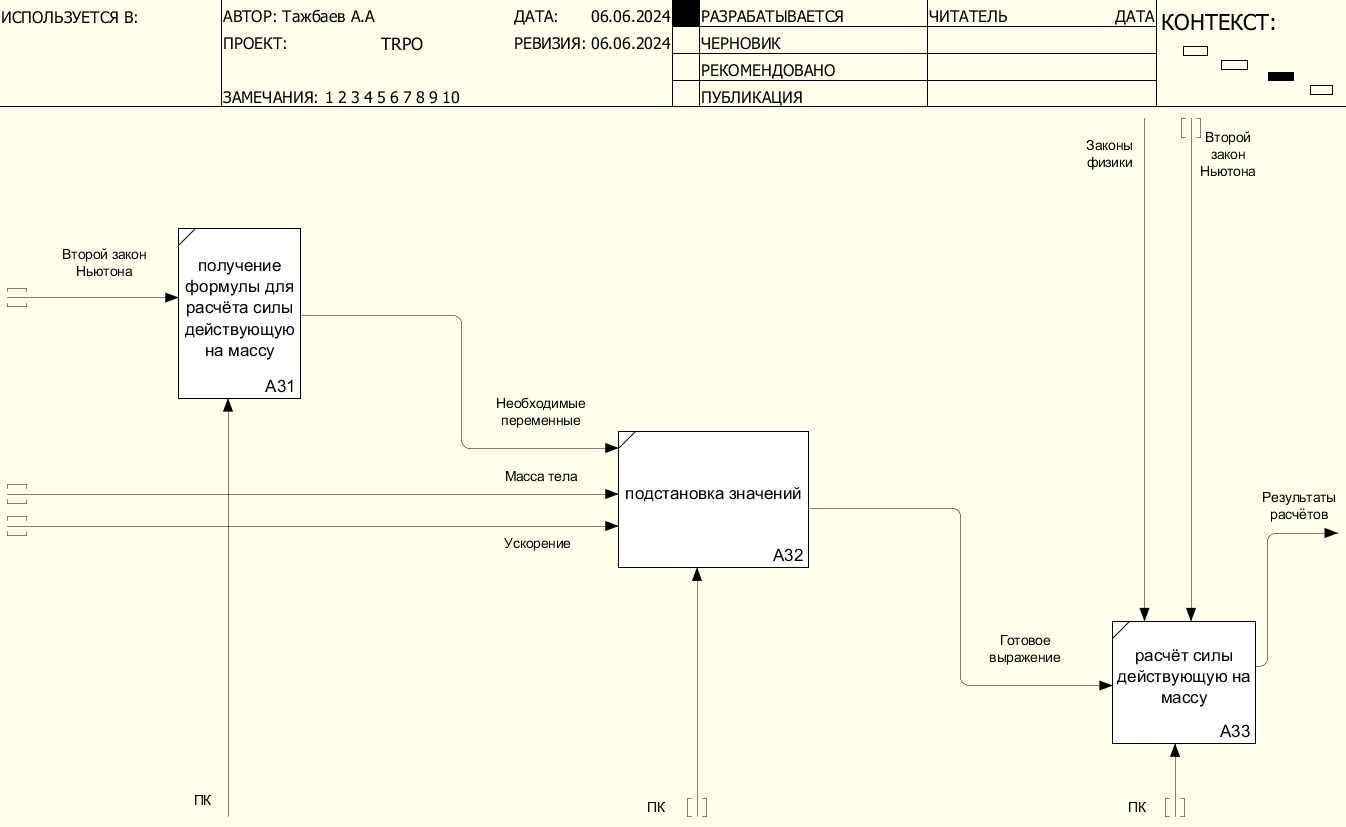


Рисунок 3 – Диаграмма декомпозиции IDEF0 второго уровня

Все функциональные диаграммы будут использованы при разработке программного обеспечения для автоматизации расчётов в классической механике.

# Разработка и тестирование программного продукта

## Обоснование программы средств реализации

Для реализации системы по разработке программного обеспечения для автоматизации расчётов в классической механике, выбран язык программирования C# и среда разработки Visual Studio.

Язык программирования C# (C Sharp) был создан в 2000 году компанией Microsoft и с тех пор стал одним из наиболее популярных языков программирования благодаря своей мощности, гибкости и простоте в использовании. Он является объектно-ориентированным языком, что означает, что код организуется в виде объектов, что упрощает разработку сложных программ и повышает их структурную чистоту.

Одним из ключевых преимуществ C# является его строгая типизация. Это означает, что все переменные должны быть определены с определенным типом данных, что позволяет выявлять ошибки на этапе компиляции и обеспечивает более надежное выполнение программы. Кроме того, C# обладает механизмом обработки исключений, который позволяет разработчикам более эффективно управлять ошибками и создавать стабильные приложения.

Среда разработки Visual Studio предоставляет широкий спектр инструментов для работы с языком C#. Она включает в себя редактор кода с подсветкой синтаксиса, автодополнением и множеством других функций, которые помогают ускорить процесс разработки. Visual Studio также интегрируется с другими инструментами разработки, такими как системы контроля версий, тестирование и развертывание приложений.

.NET Core, платформа для разработки кроссплатформенных приложений на C#, позволяет создавать приложения, которые могут работать на различных операционных системах, что делает их более универсальными и доступными для большего числа пользователей.

Обширное сообщество разработчиков C# и Visual Studio предлагает множество ресурсов для обучения, поддержки и обмена опытом. Это позволяет как новичкам, так и опытным разработчикам получить помощь и решить возникающие проблемы.

В целом, выбор C# и Visual Studio для создания системы по автоматизации расчётов в классической механике обусловлен их современностью, эффективностью, удобством использования и поддержкой сообщества. Эти инструменты позволят создать высококачественное программное обеспечение для проведения сложных расчетов с высокой точностью и надежностью.

Использование языка программирования C# и среды разработки Visual Studio также обеспечивает возможность создания графического пользовательского интерфейса (GUI) для удобного взаимодействия пользователя с программой. Благодаря богатым возможностям библиотеки Windows Presentation Foundation (WPF) и инструментов дизайна в Visual Studio.

## Разработка пользовательского интерфейса

Пользовательский интерфейс (UI – User Interface) в программном обеспечении — это то, через что пользователь взаимодействует с программой. Он представляет собой набор элементов, которые позволяют пользователю управлять программой, вводить данные, просматривать информацию и взаимодействовать с функционалом программы.

Особенности пользовательского интерфейса:

1 Интуитивность: UI должен быть понятен и легок для использования даже без дополнительного обучения.

2 Эффективность: Интерфейс должен позволять пользователям выполнять задачи быстро и эффективно.

3 Привлекательный дизайн: Визуальное оформление UI должно быть приятным и привлекательным для пользователя.

4 Консистентность: Элементы интерфейса должны быть последовательными и предсказуемыми в своем поведении.

5 Отзывчивость: UI должен реагировать на действия пользователя незамедлительно, без задержек.

6 Доступность: Интерфейс должен быть доступен для всех категорий пользователей, включая людей с ограниченными возможностями.

Требования к пользовательскому интерфейсу:

1 Простота использования: UI должен быть интуитивно понятным и легким для освоения.

2 Эффективность: Интерфейс должен помогать пользователям выполнять задачи быстро и без лишних усилий.

3 Наглядность: Интерфейс должен ясно отображать информацию и обеспечивать легкое восприятие.

4 Гибкость: UI должен быть гибким и адаптивным к различным устройствам и разрешениям экрана.

5 Безопасность: Интерфейс должен обеспечивать защиту данных и конфиденциальность пользователей.

6 Согласованность: Элементы интерфейса должны быть согласованными между собой и со стандартами дизайна.

7 Обратная связь: UI должен предоставлять пользователю обратную связь о результатах его действий.

Важной частью UI является также доступность. Хороший пользовательский интерфейс должен быть доступен для всех пользователей, включая людей с ограниченными возможностями. Это означает, что UI должен быть разработан с учетом использования вспомогательных технологий, таких как скринридеры, а также предоставлять возможности для увеличения размеров шрифтов или изменения цветовой схемы.

Наконец, хороший пользовательский интерфейс должен быть эстетичным и соответствовать бренду или общему стилю продукта.

Программный интерфейс автоматизированной системы расчётов в классической механике должен удовлетворять всем перечисленным выше требованиям. Структура и компоненты программного интерфейса, разрабатываемой АС предоставлены на рисунке 4.



Рисунок 4 – Главная форма интерфейса

На главной форме расположены кнопки с названиями «Задача 1», «Задача 2», «Задача 3», «Задача 4», «Задача 5», «Задача 6» и «Задача 7». Фоновым изображением служит фотография, на которой изображены Галилео Галилей и Исаак Ньютон. Элементы формы размещены таким образом, чтобы создать гармоничный и привлекательный дизайн, который отражает тему расчёта в классической механике.

Форма для расчёта периода колебания математического маятника задачи № 1 расположена на рисунке 5.

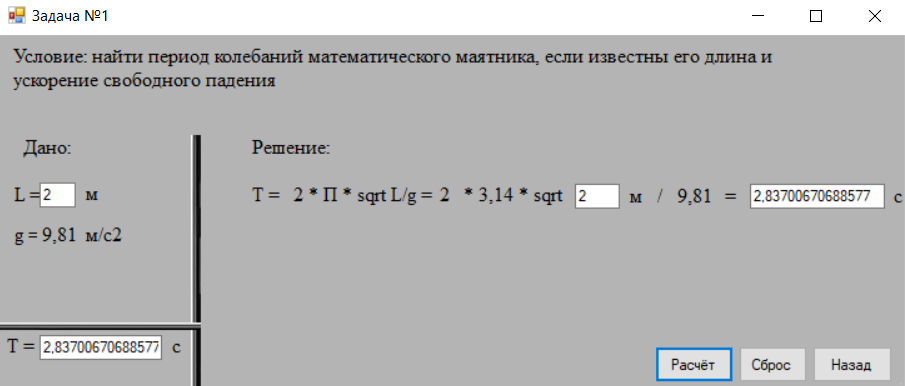


Рисунок 5 – Форма для расчёта периода колебания маятника

На данной форме представлен функционал для расчёта периода колебания математического маятника. Пользователю предоставляется возможность ввести значение длины маятника L и получить результат расчёта периода колебания. Пользователь вводит значение длины маятника в TextBox, затем нажимает кнопку «Рассчитать». После этого программа использует формулу T = 2π√(L/g) для вычисления периода колебания, где T - период колебания, L - длина маятника, а g - ускорение свободного падения.

Форма расчёта силы на массу задачи № 2 показана на рисунке 6.

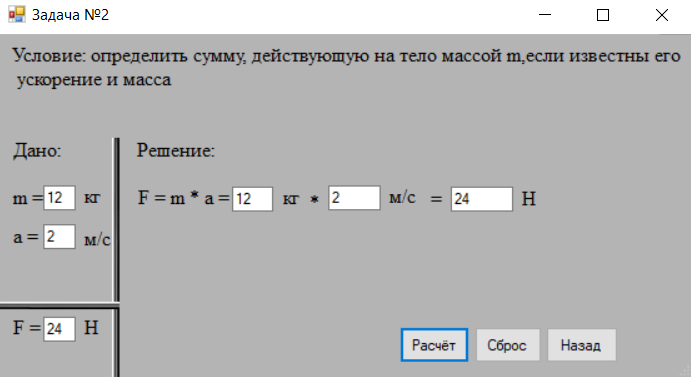


Рисунок 6 – Форма для расчёта силы, действующей на тело

Данная форма представляет собой пользовательский интерфейс, разработанный для удобного расчёта силы, воздействующей на объект определенной массы. Она состоит из нескольких элементов, каждый из которых выполняет свою функцию. Пользователю необходимо ввести свои значения в поле m и а, после чего производится расчёт по формуле F = m \* a, которая запускает процесс вычисления силы, действующей на тело. После расчёта программа выводит результат в TextBox. Кнопка «Сброс», предназначенная для сброса всех введенных пользователем значений массы и ускорения. При нажатии на эту кнопку все поля ввода очищаются, что позволяет начать новый расчёт с чистого листа. Кнопка «Назад», предоставляющая пользователю возможность вернуться к главной форме.

Форма расчёта кинетической энергии показана на рисунке 7.

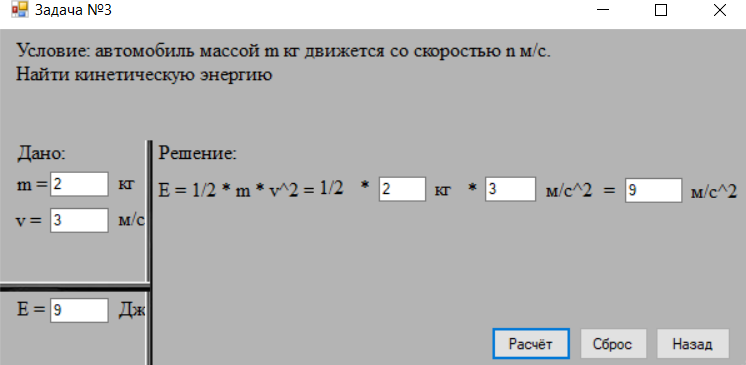


Рисунок 7 – Форма для расчёта кинетической энергии

Данная форма представляет собой интерактивный пользовательский интерфейс для расчета кинетической энергии. На форме имеются два текстовых поля для ввода значений: массы (m) и скорости (v). Пользователь может ввести необходимые числовые значения в эти поля для дальнейшего расчета. После ввода значений пользователь может нажать кнопку «Рассчитать», что приведет к выполнению расчета кинетической энергии по формуле E = 1/2 m v^2. Результат расчета отобразится на форме, позволяя пользователю увидеть полученное значение кинетической энергии. Кроме того, на форме присутствует кнопка «Назад», которая позволяет пользователю вернуться к главной форме. Также на форме имеется кнопка «Сброс», предназначенная для очистки всех введенных пользователем значений массы и скорости. При нажатии на эту кнопку все поля ввода обнуляются, позволяя начать новый расчет с чистого листа.

Форма для расчёта кинетической энергии тела задачи № 4 расположена на рисунке 8.

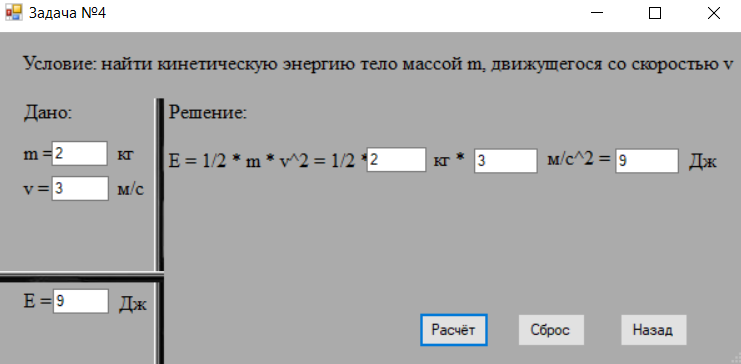


Рисунок 8 - Форма для расчёта кинетической энергии

Данная форма представляет собой пользовательский интерфейс для расчета кинетической энергии тела. На форме также присутствуют два текстовых поля для ввода значений: m и v. Пользователь может ввести необходимые числовые значения в эти поля для расчёта. После ввода значений пользователь может нажать кнопку «Рассчитать», что приведет к выполнению расчета кинетической энергии по формуле E = 1/2 m v^2. Результат расчета отобразится на форме. Кроме того, на форме присутствует кнопка «Назад», которая позволяет пользователю вернуться к предыдущему экрану. Также на форме имеется кнопка «Сброс», предназначенная для очистки всех введенных пользователем значений массы и скорости. При нажатии на эту кнопку все поля ввода обнуляются, позволяя начать новый расчёт.

Форма для вычисления потенциальной энергии задачи № 9 показана на рисунке 10.

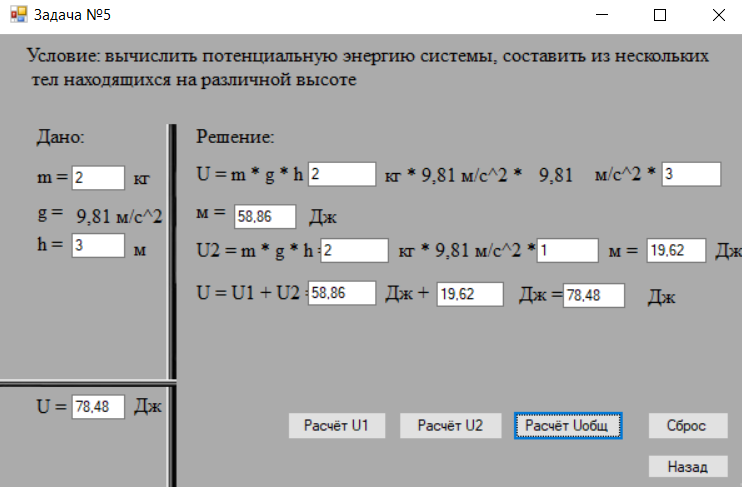


Рисунок 9 – Форма для расчёта потенциальной энергии

Данная форма визуальной студии представляет собой интерактивный пользовательский интерфейс для расчёта потенциальной энергии и их суммирования. На форме присутствуют следующие элементы:

1 Поле ввода массы (m): Пользователь может ввести значение массы тела.

2 Поле ввода высоты (h): Пользователь может указать значение высоты, на которой находится тело.

3 Кнопка «U1»: при нажатии на эту кнопку происходит расчёт потенциальной энергии (U1) по формуле U1 = m \* g \* h, где g - ускорение свободного падения.

4 Поле вывода результата U1: здесь отображается результат расчета потенциальной энергии U1.

После расчёта U1 пользователь может:

5 Поле ввода собственного значения U2: здесь пользователь может ввести свое значение потенциальной энергии.

6 Кнопка «U2»: при нажатии на эту кнопку происходит расчёт пользовательской потенциальной энергии (U2).

7 Поле вывода результата U2: здесь отображается результат расчёта пользовательской потенциальной энергии U2.

После расчета U2 пользователь может:

8 Поле для суммирования результатов (U общ): здесь пользователь может ввести полученные значения U1 и U2 для их суммирования.

9 Текстовое поле для вывода результата суммарной потенциальной энергии: здесь отображается результат сложения значений U1 и U2.

Кроме того, на форме присутствуют следующие элементы управления:

Кнопка «Назад»: при нажатии на эту кнопку пользователь может вернуться к предыдущему экрану или шагу, что обеспечивает удобство использования формы.

Кнопка «Сброс»: при нажатии на эту кнопку все поля ввода будут очищены, что позволяет начать новый расчет с чистого листа.

Форма для расчёта результативной силы задачи № 6 показана на рисунке 10.

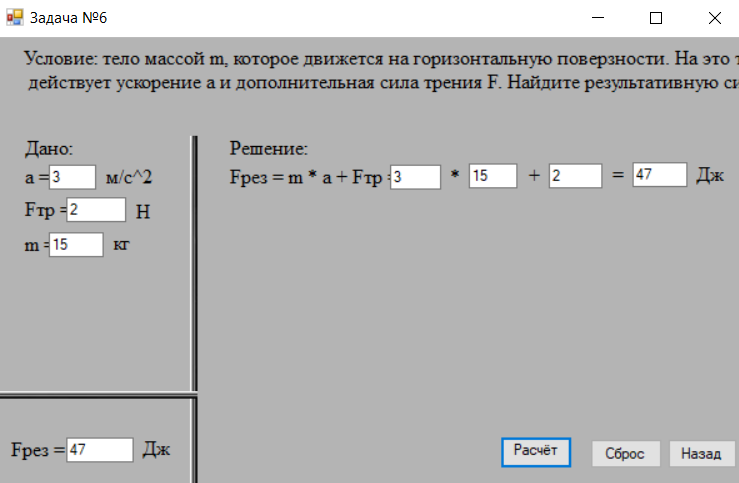


Рисунок 10 – Форма для расчёта результативной силы

Данная форма представляет собой интерактивный пользовательский интерфейс для расчета результативной силы. На форме присутствуют следующие элементы:

1 Поле ввода значения ускорения (a): Пользователь может ввести значение ускорения, с которым движется тело.

2 Поле ввода значения силы трения (Fтр): Пользователь может указать значение силы трения, действующей на тело.

3 Поле ввода значения массы тела (m): Пользователь может ввести значение массы тела.

4 Кнопка «Рассчитать»: при нажатии на эту кнопку происходит расчёт результативной силы (Fрез) по формуле Fрез = m \* a + Fтр, где m - масса тела, a - ускорение и Fтр - сила трения.

5 Поле вывода результата Fрез: здесь отображается результат расчёта результативной силы.

Кроме того, на форме присутствуют следующие элементы управления:

Кнопка «Назад»: при нажатии на эту кнопку пользователь может вернуться к предыдущему экрану или шагу, что обеспечивает удобство использования формы.

Кнопка «Сброс»: при нажатии на эту кнопку все поля ввода будут очищены, что позволяет начать новый расчёт с чистого листа.

Форма для определения ускорения тела задачи № 7 показана на рисунке 11.

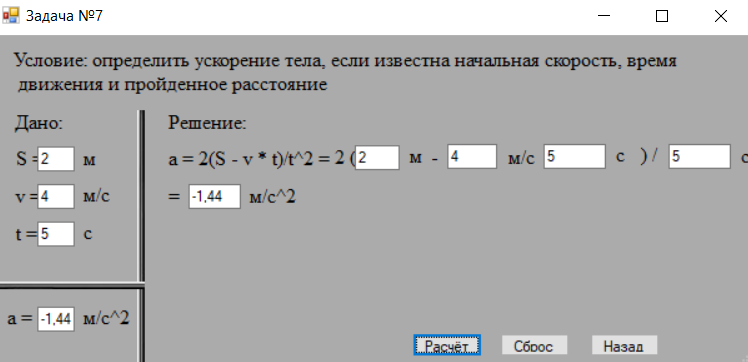


Рисунок 11 – Форма для расчёта ускорения

Данная форма представляет собой интерактивный пользовательский интерфейс для расчёта ускорения тела. На форме присутствуют следующие элементы:

1 Поле ввода значения перемещения (s): здесь пользователь может ввести значение перемещения тела.

2 Поле ввода значения начальной скорости (v): в это поле пользователь вводит значение начальной скорости тела.

3 Поле ввода значения времени (t): Пользователь указывает здесь значение времени, за которое произошло движение тела.

4 Кнопка «Рассчитать»: при нажатии на эту кнопку происходит расчёт ускорения по формуле a = 2(s - vt)/t^2, где s - перемещение, v - начальная скорость, t - время.

5 Поле вывода результата ускорения (a): здесь отображается результат расчёта ускорения тела.

Кроме того, на форме присутствуют следующие элементы управления: кнопка «Назад»: при нажатии на эту кнопку пользователь может вернуться к предыдущему экрану или шагу, что обеспечивает удобство использования формы. Кнопка «Сброс»: при нажатии на эту кнопку все поля ввода будут очищены, что позволяет начать новый расчёт с чистого листа.

## Алгоритмизация и программирование

Алгоритм — это последовательность шагов, предназначенная для решения определенной задачи. Он представляет собой точное и конкретное описание того, как выполнять определенную операцию или решать определенную проблему.

Свойство алгоритма включает в себя следующие характеристики:

1 Дискретность: алгоритм должен состоять из конечного числа шагов, каждый из которых является четко определенным и дискретным.

2 Определённость: каждый шаг алгоритма должен быть однозначно понятен и выполним.

3 Входные данные: алгоритм должен иметь определённые входные данные, на основе которых он выполняет свои шаги.

4 Выходные данные: после завершения работы алгоритм должен выдавать определённый результат или выходные данные.

5 Эффективность: алгоритм должен решать задачу за разумное время и с разумными ресурсами.

Алгоритмы чаще всего представляются в виде блок схем по стандарту ГОСТ 19.701 – 90 «Единая система программной документации. Схемы алгоритмов, программ данных и систем условные обозначения и правила выполнения»

Задача №1: найти период колебаний математического маятника, если известны его длина и ускорение свободного падения.

Блок-схема алгоритма решения задачи № 1 показана на рисунке 12.

Start

L,g

T = 2\*П\*sqrt L/g

T

end

Рисунок 12 – Блок-схема задачи для расчёта периода колебаний

Задача №2: определить силу, действующую на тело массой m, если известны его ускорение и масса.

Блок-схема алгоритма решения задачи № 2 показана на рисунке 13.

Start

m,a

F=m\*a

F

end

Рисунок 13 – Блок-схема задачи для расчёта силы на массу

Задача №3: автомобиль массой m движется со скоростью v. Найдите кинетическую энергию автомобиля.

Блок-схема алгоритма решения задачи № 3 показана на рисунке 14.

Start

m,v

E=1/2 \* m \*v^2

E

end

Рисунок 13 – Блок-схема задачи для расчёта кинетической энергии

В АС расчётов в классической механике программно реализованы следующие методы, расположенные на формах со второй по седьмую:

 1) button1\_Click: Этот метод вызывается при нажатии на кнопку button1. Внутри метода происходит вычисление значения по заданной формуле и отображение результата на форме задачи №1.

 2) textBox1\_TextChanged, textBox3\_TextChanged, textBox2\_TextChanged, textBox4\_TextChanged: Эти методы вызываются при изменении текста в соответствующих текстовых полях textBox1, textBox3, textBox2 и textBox4. Они обновляют содержимое других текстовых полей (textBox5, textBox6, textBox7, textBox8) при изменении значений, вводимых пользователем.

 3) button2\_Click: Этот метод вызывается при нажатии на кнопку button2. Внутри метода происходит очистка всех текстовых полей на форме.

 4) button3\_Click: Этот метод вызывается при нажатии на кнопку button3. Внутри метода происходит скрытие текущей формы и переход к главной форме.

## 2.5 Тестирование

# Рекомендации по внедрению, эксплуатации и сопровождению программного продукта

## Руководство пользователя

Обзор возможностей программы

Программа для решения задач для классической физики — это программа, которая помогает решать задачи при помощи формул.

Программы для расчётов в классической механике предназначена для проведения численных расчетов, анализа физических явлений и применения математических моделей для решения задач классической физики. Они помогают проводить сложные вычисления, моделировать движение тел и эффективно решать различные задачи механики.

Основными выгодами от использования расчёта классической механики являются:

1 Увеличение точности расчетов за счет использования сложных математических моделей.

2 Ускорение процесса анализа и решения задач благодаря автоматизации вычислений.

3 Возможность проведения сложных и объемных расчетов, которые были бы трудоемкими или невозможными для выполнения вручную.

Системные требования

Для стабильной и эффективной работы расчётов в классической физике рекомендуется использовать следующую конфигурацию:

Частота процессора (CPU): 2.5 GHz

Количество ядер процессора (CPU): 8

Объем оперативной памяти (RAM): 16 GB

Объем свободного места на диске (HDD): 3 GB

Операционная система (OS): Windows 11, Windows 10.

Браузер: Google Chrome, Mozilla Firefox, Microsoft Edge

Начало работы

Данный раздел поможет вам быстро установить, настроить и начать работать с расчётами классической физики.

Перед началом работы, пожалуйста, ознакомьтесь с [системными требованиями](#a9eb003e-8704-4d8f-8d92-8f59944ba6d6) и [лицензионным соглашением](#0343637b-d224-4986-a470-e45222a3137f).

Основные понятия и термины

Перед началом работы в расчётах молекулярной физики рекомендуем ознакомиться с основными понятиями и терминами:

Понятие 1

Классическая механика — вид механики (раздела физики, изучающего законы изменения положений тел в пространстве со временем и причины, его вызывающие), основанный на законах Ньютона и принципе относительности Галилея. Поэтому её часто называют «ньютоновой механикой».

Понятие 2

Скорость — векторная физическая величина, характеризующая быстроту перемещения и направление движения материальной точки относительно выбранной системы отсчёта.

Понятие 3

Масса — скалярная физическая величина, определяющая инерционные и гравитационные свойства тел в ситуациях, когда их скорость намного меньше скорости света

Понятие 4

Компьютер - компьютер представляет собой электронное устройство, которое работает с информацией и данными.

Понятие 5

Программа - последовательность машинных команд, предназначенная для достижения конкретного результата.

Понятие 6

Dr. Explain — это приложение для быстрого создания файлов справки (help-файлов), справочных систем, on-line руководств пользователя, пособий и технической документации к программному обеспечению и техническим системам.

Понятие 7

Интегрированная среда разработки Visual Studio является творческой стартовой площадкой, которую можно использовать для редактирования, отладки и сборки кода, а также для публикации приложения.

Понятие 8

Ramus — программа для построения визуальных диаграмм, используемых для наглядного отображения различных бизнес-процессов.

Понятие 9

Программный код — набор инструкций для компьютера. Его пишут на языке программирования сами разработчики или генерируют автоматически. С помощью кода создают программы: отдают компьютеру команды, которые он выполняет.

Понятие 10

Документация — совокупность документов, посвященных какому-либо вопросу (задаче, проекту, изделию и др.). Документирование — процесс отбора, классификации, использования и распространения документов.

Установка

Для установки Visual studio, пожалуйста, загрузите дистрибутива последней версии 5.7.1, доступный по адресу <https://learn.microsoft.com/ru-ru/visualstudio/releases/2019/release-notes-v16.10>

Перед установкой ознакомьтесь с [системными требованиями](#a9eb003e-8704-4d8f-8d92-8f59944ba6d6) и [лицензионным соглашением](#0343637b-d224-4986-a470-e45222a3137f).

В процессе установки, пожалуйста, убедитесь, что системные требования соответствует программному продукту.

Настройка

Для начала работы в Visual Studio рекомендуем предварительно выполнить следующие настройки окружения:

1 Установить Visual Studio

2 Настроить дополнительные расширения и плагины для Visual Studio в зависимости от потребностей разработки

3 Создать проект или открыть существующий проект для начала работы над разработкой программного обеспечения.

Запуск

Для запуска расчёта классической механики нажмите на ярлык программы <Visual Studio> в меню Пуск либо наберите в командной строке <Visual Studio>

При первом запуске программы <Открывается главное окно программы с интерфейсом>

Пользовательский интерфейс

Этот раздел описывает основные элементы пользовательского интерфейса расчётов в классической физике 1.0: основных режимов работы, предназначение окон и экранов, доступные операции.

Главное окно программы

Главное окно программы расчёта классической физики позволяет выполнять следующие операции:

Открыть задачи классической физике

Расчёты по формулам



Настройки программы

Для начала работы в расчёте классической физики следующие настройки окружения:

1. Создать проект в среде Visual Studio;

2. Распаковать файлы курсовой в папке проекта;

3. Завершить проект;

Режимы работы с задачами по молекулярной физики

Пользовательский интерфейс расчётов в классической физике обеспечивает работу в нескольких режимах: Программа работает только в режиме пользователя.

Нет ограничений.

Пользовательский режим

Программа работает в пользовательском режиме, в котором доступен весь функционал.

Работа с кнопками

Данный раздел описывает работу с кнопками программы

В частности, рассматриваются наиболее частые операции:

1 Кнопка «Задача 1-7», при нажатии на неё откроется новое окно, в котором можно будет решить задачу по формуле;

2 Кнопка «Назад», при нажатии на неё переносит на главную форму;

3 В окнах «Задача 1-7», есть кнопки «Сброс» «Решение» «Назад»

4 Кнопка «Сброс» после нажатия этой кнопки сбросится все введённые данные и решение

Горячие клавиши

Следующий раздел содержит все сочетания клавиш и способы управления при помощи мыши, поддерживаемые в программном продукте.

Общие

Ctrl+N — создать новый проект.

Ctrl+O — открыть проект.

Ctrl+S — сохранить открытый проект.

Редактирование

Alt+BackSpace — отменить.

Shift+Delete — вырезать.

Shift+Insert — вставить.

Ctrl+C — копировать.

Ctrl+V — вставить.

Ctrl+Y — повторить.

Ctrl+A — выбрать все.

Ctrl+Z — отменить.

Примеры использования

В данном разделе собраны примеры реального использования расчёта классической механики, демонстрирующие применения продукта в различных отраслях.

Работа с задачами

Задача

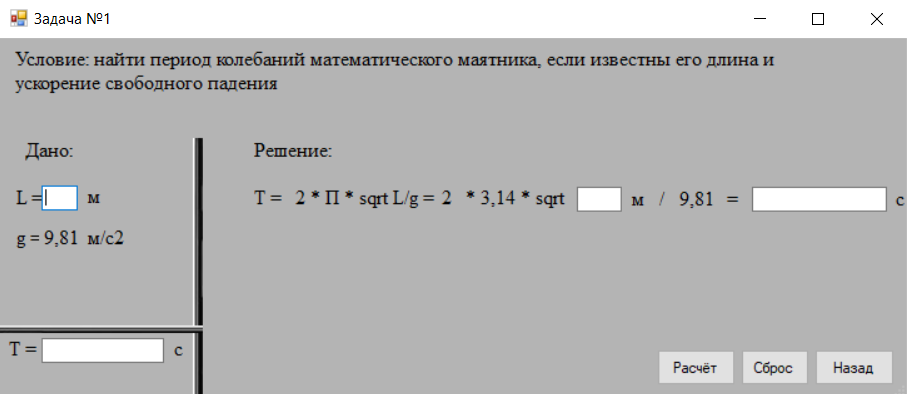
Рассмотрим на примере задач. Найти период колебаний математического маятника, если известны его длина и ускорение свободного падения

Проблема

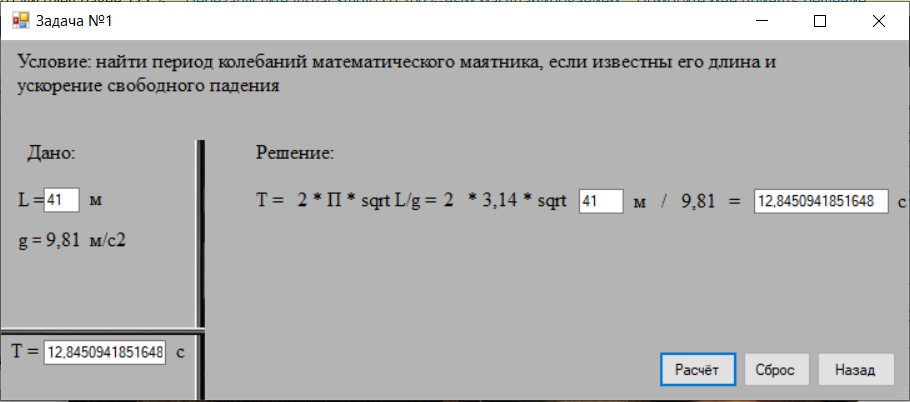
Если пользователь вводит некорректные данные, например символы букв, «-», «\_2», «!», «?» и т.д. То строки с данными не будут выводить результат.

Результат

После того как пользователь заполнил все данные поля данными



Ему следует нажать кнопку «Рассчитать», после чего программ решит данную задачу по формуле.



Вывод:

На примере этой задачи можно сделать вывод что программа способна решать задачи в классической механике по формулам

Устранение типовых проблем

|  |  |
| --- | --- |
|  | |
|  | Описание проблемы ввод неправильные данных  Решение: перезапустить программу. И ввести правильные данные задачи |
|  | |
|  | |
|  | Описание проблемы зависание программы  Решение: перезапустить программу Visual Studio 2019 |
|  | |
|  | |
|  | Описание проблемы несовместимость операционной системы  Решение: установить новую версию Windows  Необходимые операционные системы: Windows 11, Windows 10. |
|  | |

Частые вопросы (FAQ)

|  |  |
| --- | --- |
|  | |
|  | Как открыть программу?  Ответ: чтобы открыть программу, нужно запустить Visual Studio 2019 |
|  | |
|  | |
|  | Как скачать программу?  Ответ: найти в браузере Visual Studio 2019 community. |
|  | |

Если вы не нашли ответа на свой вопрос, пожалуйста, [свяжитесь с нами](file:///C:\Users\ACER\Downloads\Telegram%20Desktop\c3676b4db61411f8fb13557a8f8e87ba.docx#7f77a01d-3465-466f-ba5d-ec49887442b5).

**Контактная информация**

Расчёты молекулярной физики разрабатывается и поддерживается компанией **22-ИСП2**, являющейся правообладателем.

**Техническая поддержка**

Вы можете направить вопросы по функциональности программы расчётов молекулярной физики следующими способами:

* Email: pakzuc@gmail.com
* Телефон: +7 995 276 10-94
* Мессенджеры: ВК: @dynastysiemens
* Форма обратной связи: Заочная.

## Планы внедрения и сопровождения

# Заключение

2строки

На одну страницу, подвести итоги проделанной работы

# Список использованных источников

1. Работы студенческие. Общие требования и правила оформления. СТО 02069024.101 – 2015. – Оренбург: Изд-во ОГУ, 2015. – 89 с.
2. <https://ru.wikipedia.org/wiki/Автоматизированная_система> - общие данные о автоматизированной системе
3. <https://djvu.online/file/V2M89HmQhPoeI> - техника вычислений в классической механике
4. <https://www.researchgate.net/> - научные статьи, публикации и исследования по классической механике
5. <https://ieeexplore.ieee.org/> - множество научных статей и журналов по физике и инженерии, которые могут быть полезны при написании курсовой работы
6. <https://arxiv.org/> - предпечатные версии научных статей по физике, включая темы классической механики.
7. <https://nauchniestati.ru/spravka/klassicheskaya-mehanika/> - основные принципы, описание законов физики и классификация движений в классической механике
8. <https://zaochnik.ru/blog/sovety-v-reshenii-zadach-po-klassicheskoj-mexanike-soxranyaj-i-polzujsya/> - примеры решения задач в классической механике

# Приложение А

(обязательное)

**Текст программы**