

ОЛИМПИАДА ШКОЛЬНИКОВ «ШАГ В БУДУЩЕЕ»

НАУЧНО-ОБРАЗОВАТЕЛЬНОЕ СОРЕВНОВАНИЕ «ШАГ В БУДУЩЕЕ, МОСКВА»

ШМ0607

*регистрационный номер*

Информатика и системы управления (ИУ)

*название факультета*

Программное обеспечение ЭВМ и информационные технологии (ИУ-7)

*название кафедры*

Разработка обучающей программы с возможностью закрепления и проверки  
знаний на примере раздела физики

*название работы*

Автор:

Иванов Всеволод Алексеевич

*фамилия, имя, отчество*

ГБОУ Лицей № 1581 города Москвы, 10 Г

*наименование учебного заведения, класс*

Научный руководитель:

Литвин Елена Николаевна


*фамилия, имя, отчество*

ГБОУ Лицей № 1581 города Москвы

*место работы*

к. ф.-м. н., учитель информатики

*звание, должность*

  
*подпись научного руководителя*

## СОДЕРЖАНИЕ:

Рекламный лист _____	1 стр.
Введение _____	2 стр.
Актуальность _____	2 стр.
Цель и задачи _____	3 стр.
Изложение алгоритма решения задачи _____	3 стр.
Возможности улучшения продукта _____	10 стр.
Проблемы, возникшие в процессе разработки _____	10 стр.
Возможность применения _____	10 стр.
Заключение _____	11 стр.
Системные требования _____	11 стр.
Используемые средства _____	12 стр.
Приложение 1 _____	13 стр.

## **Рекламный лист**

В наше время большую роль в образовательной системе играют информационные технологии. Настоящим представляется проект, в котором посредством информационных технологий была создана программа, позволяющая пользователю самостоятельно изучать теоретический материала, закреплять полученные знания.

Интерфейс программы состоит из окон, позволяющих пользователю просматривать теоретический материал, решать задачи и тесты. Главным отличием от аналогов является наличие редактора материала, позволяющего любому пользователю легко и удобно редактировать контент. Данная функция создаёт возможность любому педагогу легко создать своё пособие, так и студенту или школьнику вести свой уникальный конспект.

Для улучшения навигации по материалу предоставлена функция создания ссылок на теоретический материал, связанный по смыслу с редактируемым материалом

Созданная в программе структура хранения и представления данных позволит использовать её практически для любой учебной дисциплины.

Программа компилирована в исполняемый файл, что позволяет использовать её на любом компьютере с операционной системой Windows. Системные требования минимальны.

## **Введение**

Современный выбор инструментов позволил создать широчайший ассортимент программных продуктов в сфере образования. Существуют электронные задачки, сетевые учебники с внушительными наборами библиотек. Что нового можно предложить на этом, казалось бы, освоенном пространстве?

## **Актуальность**

Образование всегда являлось одной из самых актуальных тем в жизни человека. На данный момент в образовательную систему активно внедряются информационные технологии (например, электронный дневник МРКО, ставший заменой бумажному аналогу и сайты дистанционной подготовки, даже экзамены на знание ПДД давно уже проходят в формате диалога с соответствующей программой)

Данный проект предлагает новый взгляд на программу для обучения. Новизна состоит прежде всего в том, что формирование и редактирование контента полностью находится в руках пользователя.

Большинство образовательного ПО содержит только теоретический материал, либо практические тесты. Данная программа создаёт возможность тесно связывать теоретический материал и задания для проверки знаний пользователя. Данная функция позволит пользователю, изучившему теоретический материал, сразу проверить свои знания с помощью задач или тестов. И наоборот, в случае непонимания пользователем тестирующего материала существует возможность восполнить «пробелы» в знаниях, изучив теоретический материал.

## **Цель и задачи**

Целью данного проекта является создание программы, позволяющей просматривать и изменять учебный материал в доступной для любого пользователя форме.

В программе предполагается использование графического интерфейса, изображений и баз данных

В процессе разработки программы были поставлены следующие задачи:

- 1) Изучение общей структуры теоретического материала, задач, тестов для создания обобщённой формы их представления
- 2) Создание баз данных для хранения учебного материала
- 3) Создание графического интерфейса
- 4) Создание редактора баз данных для добавления, изменения и удаления учебного материала
- 5) Создание алгоритма отображения учебного материала
- 6) Добавление в базы данных учебного материала
- 7) Компиляция программы в .exe файл

## **Изложение алгоритма решения задачи**

Изучение общей структуры теоретического материала, задач, тестов для создания обобщённой формы их представления

Для представления теоретического материала, в основном, используется текст, комбинированный с изображениями (изображения находятся непосредственно в самом тексте для удобства его восприятия). Следовательно, для создания статьи необходимо использовать текст и изображения с указанием их места в тексте.

Для представления задач, в основном, используется условие, ответ и изображение (в большинстве случаев изображение отсутствует, либо используется одно изображение, положение которого не влияет на суть условия). Следовательно, для создания задачи необходимо указать условие задачи и, отдельно, указание изображения(ний). Аналогично представляется и решение задач. В отдельных случаях требуется использование изображений в условии или решении. Для этого изображения могут быть указаны в условии или решении аналогично изображениям в теоретическом материале.

Для представления вопросов из тестов используется текст и изображения. Так же, как и в задачах, изображения не требуют привязки к тексту. Сами вопросы делятся на вопросы типа А, в которых предоставляется несколько вариантов ответа, из которых нужно выбрать один правильный, и вопросы типа В, в которых нужно дать краткий ответ. Существуют также и вопросы типа С, для которых требуются развёрнутый ответ, но ответы на такие вопросы может проверять только педагог. В некоторых случаях, в вопросах типа А вариантами ответа являются изображения (в случаях, когда вариантами ответа являются графики и т.п.). Следовательно, для создания вопроса из теста необходимо определить его условие, тип, изображения, связанные с условием, варианты ответа или вариант, при этом должна существовать возможность использовать в качестве варианта ответа изображение.

Для трёх вышеперечисленных типов образовательного материала также требуется указание темы, к которой данный материал принадлежит и указание теоретического материала, связанного с данным материалом.

#### Создание баз данных для хранения учебного материала

Для хранения материала используются базы данных MySQL. Взаимодействие с базами данных осуществляется за счёт встроенной в Python библиотекой sqlite3

На основе созданного представление о структуре учебного материала, были созданы базы данных, хранящие:

- 1) Теоретический материал; Для его хранения в базе данных используются поля: названия предмета, раздела и темы, к которым относятся данная статья, название статьи, номер в базе данных, текст, и ссылки на другие статьи, связанные по смыслу с данной статьёй
- 2) Задачи; Для их хранения в базе данных используются поля: названия предмета, раздела и темы, к которым относятся данная задача, номер в базе данных, её название (если название не указано пользователем, то используется её номер в базе данных), условие задачи, названия изображений, используемых в условии, ответ, решение задачи, названия изображений, используемых в решении, ссылки на статьи, связанные с задачей. Решение, ссылки на статьи, изображения, связанные с условием и решением не обязательны для заполнения пользователем

- 3) Тесты; Для хранения тестов в базе данных, тест разбивается на отдельные вопросы, для хранения которых используются поля: номер в базе данных, названия предмета, раздела и темы, к которым относятся данный вопрос, номер варианта, номер вопроса, тип вопроса (вопрос типа А (с выбором из нескольких вариантов ответа) или типа В (с кратким ответом)), 4 варианта ответа (для вопроса типа В используется только одно из 4 полей), изображения, используемые в вопросе. Для хранения правильного варианта ответа в вопросах типа А используется первое поле.

### Создание графического интерфейса

Для создания графического интерфейса была использована встроенная графическая библиотека tkinter. В программе были созданы 10 классов подпрограмм, отвечающих за создание окон, используемых для представления материала. Данные классы при помощи виджетов tkinter создают окна меню, выбора темы, просмотра теории, задач, тестов. Также используются окна добавления нового образовательного материала. Виджеты Button и OptionMenu далее по тексту фигурируют как «кнопки» и «меню».

Для навигации в программе были созданы классы MainPage, ChooseTheme и AddTheme, позволяющие пользователю получить доступ к интересующем его материалам.

Класс MainPage (рис.1, Приложения №1) служит для создания главного меню, содержащее кнопки при нажатии на которые пользователь переходит в класс ChooseTheme (для просмотра материала) или AddThem (для редактирования)

Класс ChooseTheme (рис.2, Приложения №1) используется для выбора просматриваемой темы, интересующей пользователя. Программа составляет список всех предметов, разделов, тем и отображает их при помощи меню. После выбора темы пользователь переходит в класс TeoryWindow, TaskWindow или TestWindow.

Класс AddThem (рис.3, Приложения №1) используется для выбора редактируемой темы, интересующей пользователя. Программа составляет список всех предметов, разделов и тем и отображает их при помощи меню. В меню также присутствует пункт «Добавить». С помощью данной функции

пользователь может добавлять новый предмет, раздел или тему, если материала по ним в базе данных ещё нет. После выбора или добавления новой темы пользователь переходит в класс AddTeory, AddTask или AddTask.

### Создание редактора баз данных для добавления, изменения и удаления учебного материала

Для добавления и редактирования теоретического материала требуется создание редактора текста. Для реализации редактора теоретического материала используется класс AddTeory (рис.4, Приложения №1). Для удобства редактирования и последующего отображения существует возможность деления темы на отдельные статьи. В данном классе для редактирования текста статей используется виджет Text (поле ввода), в который из базы данных загружаются текстовые данные. Так как требуется использование изображений, привязанных к определённым отрывкам текста, простым в реализации и несложным в освоении является указание имён изображений внутри текста. Для добавления изображения в статью пользователь должен указать его имя в фигурных скобках (пример: {picture.gif}) в нужном ему месте в тексте.

Для реализации редактора задач используется класс AddTask (Приложение №6). В данном классе для редактирования условия и решения задачи используются поля ввода. Так как изображения в условии и решении не требуют привязки к тексту их можно указать отдельно от условия и решения. Для добавления изображений, связанных с условием и решением задачи используются поля ввода (Entry). Для добавления изображений в данных полях нужно указать имена изображений, разделяя их пробелами (пример: picture1.gif picture2.gif). В случае если изображение требуется вставить в определённое место текста, можно использовать метод, используемый для добавления изображений в теоретический материал.

Для реализации редактора тестов используется класс AddTest (рис.8, Приложения №1). В данном классе для редактирования вопросов и добавления изображений используется поле ввода. Аналогично изображениям в задачах, пользователь может добавлять изображения, привязанные, и не привязанные к определённому месту в тексте. В редакторе пользователь может выбрать тип теста. В случае выбора вопроса типа А пользователь может ввести от 2-х до 4-х вариантов ответа на вопрос. Ответом может являться и изображение (в таком случае должно быть

указанно его имя, например: {picture.gif}). В случае выбора вопроса типа В пользователь может указать ответ на вопрос.

Для навигации среди материала в классах AddTheory и AddTask используется меню, содержащее список всех статей или задач по данной теме. Для перехода на другую статью или задачу нужно выбрать её название в меню. Для добавления новой статьи или задачи в меню нужно выбрать пункт «Добавить». Для редактирования названия используется виджет Entry (однострочное поле ввода).

Для навигации среди тестов в классе AddTest используются меню, содержащее список всех вариантов тестов по данной теме и список всех вопросов из выбранного варианта. Для добавления нового варианта нужно выбрать «Добавить» среди списка вариантов. Для добавления нового вопроса нужно выбрать «Добавить А» или «Добавить В» (в зависимости от типа вопроса) в списке вопросов.

Для сохранения введённых данных используется кнопка «Сохранить». При переходе между статьями, задачами и тестами изменения не сохраняются. Для удаления используется кнопка «Удалить»

Изображения, используемые в материале должны находиться в папке pic в одной директорией с программой

Для удобства навигации и понимания материала существует возможность добавления ссылок на теорию, связанную с редактируемым материалом.

Перед сохранением материала программа проверяет корректность введённых данных и выдаёт сообщение, в случае введения некорректных данных.

### Создание алгоритма отображения учебного материала

Для отображения теоретического материала используется класс TheoryWindow (рис.5, Приложения №1). В данном классе пользователю предоставляется материал по теме, выбранной пользователем. При переходе в класс создаётся список всех статей данной темы, отображаемый при помощи меню, и, в случае перехода в этот класс из ChooseThem, отображает первую статью из списка, или отображает определённую статью, в случае если переход был осуществлён по ссылке или из этого же класса. Текст

статьи разбивается программой на обычный текст и изображения (названия изображений указываются в фигурных скобках). После текст и изображения последовательно отображаются при помощи виджетов Message (для отображения текста), и PhotoImage, Label (для отображения изображений). В случае если формат изображения не поддерживается или отсутствует в папке pic, выводится надпись «Некорректное изображение!». После статьи следует меню, содержащее список всех ссылок на другие статьи.

Для перемещения между статьями пользователь может использовать ссылки на другие статьи, указанные в меню, и кнопки «<» и «>», перемещающие пользователя на между статей темы.

Для отображения задач используется класс TaskWindow (рис.7, Приложение №1). В данном классе пользователю предоставляется задача по выбранной теме. При переходе в данный класс создаётся список всех задач по данной теме, отображаемый при помощи меню. В случае перехода в класс из класса из ChooseThem отображается первая задача из этого списка. Программа отображает изображения и условие задачи. Условие задачи отображается аналогично тексту в классе TeoryWindow. Рядом с условием отображаются изображения, введённые пользователем в отдельной строке. Под условием размещено поле ввода ответа на задачу и кнопка «Ответить». После введения ответа он сравнивается с правильным ответом и, если они совпадают, то выводится надпись «Правильно», если относительная погрешность введённого ответа менее 5%, то ответ считается правильным и выводится надпись «Правильно» и указание относительной погрешности. Если погрешность ответа более 5%, то ответ считается неправильным и надпись «Неправильно» и кнопка «Решение», при нажатии которой под условием отображается решение задачи и изображения, связанные с решением.

Для отображения тестов используются классы TestWindow и TestTotal.

В классе TestWindow пользователю предоставляется набор вопросов одного из тестов по выбранной теме. При переходе в данный класс программой случайным образом выбирается один из тестов по заданной теме. Из вопросов, включённых в выбранный тест, составляется список, отображаемый при помощи меню. Случайным образом определяется и запоминаются программой расположения вариантов ответов в вопросах типа А. Отображается первый вопрос из списка. Для отображения вопроса

используется текст вопроса и изображения, связанные с ним. Пользователю представляются вопросы для которых он должен выбрать один из предложенных ответов (вопрос типа А) или написать краткий ответ (вопрос типа В). В последнем вопросе выбранного теста представляется кнопка «Завершить тест». При нажатии данной кнопки пользователь переходит в класс TestTotal.

В классе TestTotal пользователю предоставляются результаты теста, в виде таблицы. Таблица содержит названия вопросов, ответы, данные пользователем, правильные ответы и количество баллов, полученное за данный вопрос. За правильный ответ на вопрос типа А присваивается 1 балл, 2 балла за вопрос типа 2. После таблицы подсчитываются все баллы, полученные пользователем, и выставляется оценка за тест (за 50% и менее – 2, менее 65% - 3, менее 85% - 4, более 85% - 5). При помощи отображаемых кнопок с надписями «Другой тест» и «Теория» пользователь может пройти другой тест по данной теме и изучить теоретический материал по данной теме (если такой существует).

Для перемещения между статьями, задачами и вопросами тестов можно использовать меню и кнопки «<» и «>».

Примечание: классы TestTotal и TestWindow в данный момент находится в стадии активной разработки

#### Добавление в базы данных учебного материала

Для наглядной демонстрации возможностей программы использован материал из раздела физики «Кинематика» по следующим причинам:

- 1) Физика является наглядным предметом, что позволяет часто использовать изображения для более точной передачи сути материала
- 2) В физике используются тесты, задачи и теоретический материал, что позволяет продемонстрировать все возможности программы
- 3) Кинематика – первый раздел из физики, изучаемый в школе

#### Компиляция программы в .exe файл

Для использования программы на компьютерах без установленного Python 3 программу требовалось скомпилировать в исполняемый файл. Для данной цели была использована программа cx\_Freeze.

### **Возможности улучшения продукта**

- 1) Создание сайта, основанного на данной программе (для реализации возможно использование CGI-скриптов). Таким образом значительно упрощается использование программы и создаётся большой потенциал для её дальнейшего развития.
- 2) Создание аккаунтов для пользователей. Данная функция позволит использовать функции, связанные с оценкой знаний пользователя и т.п.
- 3) Создание платформы для публикации созданных баз данных с образовательным материалом. Данная функция позволит значительно упростить доступ к базам данных, созданных другими пользователями и упростит публикация собственных. В рамках данной функции также возможно реализовать рейтинговую систему для опубликованных баз данных для предоставления пользователю наиболее качественного и полного образовательного материала. Для реализации данной функции необходима реализация аккаунтов в данной программе.

### **Проблемы, возникшие в процессе разработки**

Класс PhotoImage встроенной библиотеки tkinter предназначен для использования растровых изображений форматов .gif и .png.

Для устранения на данной проблемы на Python 3 на данный момент не существует официальных средств. Для устранения данной проблемы на Python 2 используется библиотека PIL.

### **Возможность применения**

Данная программа может быть использована школьниками и студентами для изучения учебного материала, проверки знаний путём решения задач и тестов.

За счёт функции добавления образовательного материала данная программа может быть использована педагогами для создания пособий, сборников тестов и задач.

За счёт использования обобщённой схемы представления теоретического материала, задач и тестов и за счёт использования

изображений на основе программы возможно реализовать большинство школьных предметов в рамках данной программы.

Начинающий программист сможет доработать программу дополнительными функциями, поскольку открытый код и используемый язык программирования доступен даже новичкам.

### **Заключение**

Результатом работы над данным проектом стала образовательная программа. Полученный продукт обладает преимуществом перед аналогами за счёт возможности привлечь к себе внимание преподавателей, заинтересованных в простом создании собственных пособий. Также программа создаёт возможность школьникам и студентам изучать предмет по самостоятельно собранному материалу.

### **Системные требования**

Программа представлена в виде исполняемого файла (.exe) и предназначена для работы в среде Windows.

Требования к системным параметрам минимальны.

### **Используемые средства**

Для разработки программы был использован язык Python 3.4.3 и встроенные в него библиотеки tkinter и sqlite3. Для хранения образовательного материала были использованы базы данных MySQL. Для компиляции программы в .exe файл была использована программа cx\_Freeze.

При изучении язык Python 3.4.3 и библиотек tkinter, sqlite3 были использованы сайты:

<http://younglinux.info/tkinter>

<http://john16blog.blogspot.ru/2011/03/python-sqlite3.html>

<http://ru.stackoverflow.com/questions/231251/Изображения-в-python>

<https://pythonworld.ru/osnovy/program-compilation-with-cx-freeze.html>

[http://www.lfd.uci.edu/~gohlke/pythonlibs/#cx\\_freeze](http://www.lfd.uci.edu/~gohlke/pythonlibs/#cx_freeze)

Для добавления учебного материала были использованы сайты:

<http://www.physics.ru>

<http://fizmatsite.narod.ru>

<http://www.afportal.ru/physics/test/easy>

Окна интерфейсов программы

Рисунок №1 (главное меню)

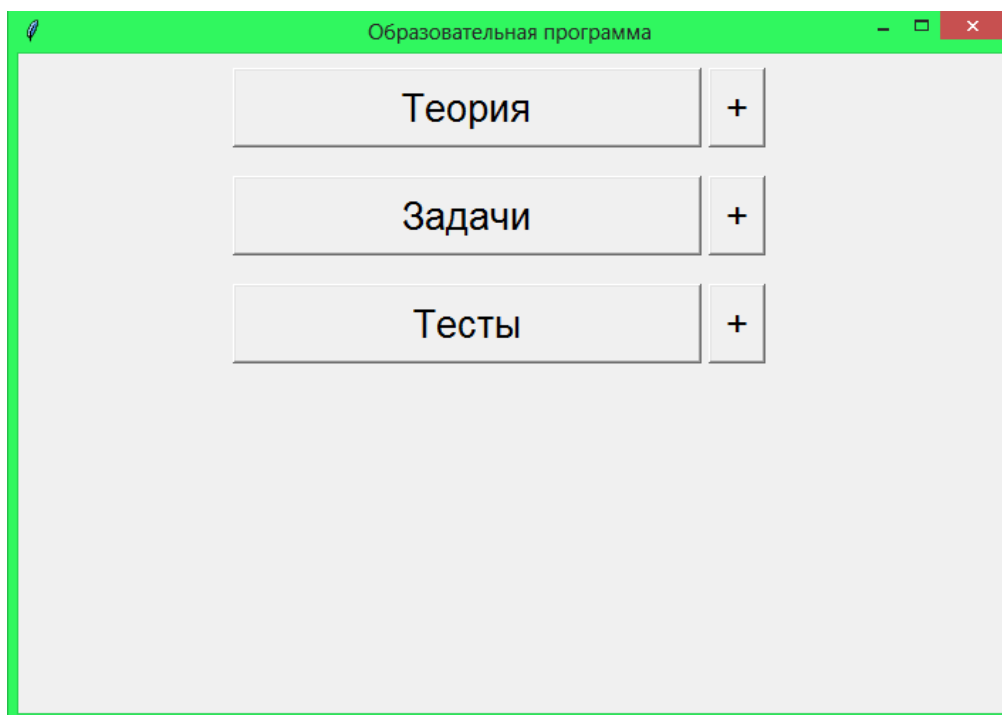


Рисунок №2 (окно выбора темы)

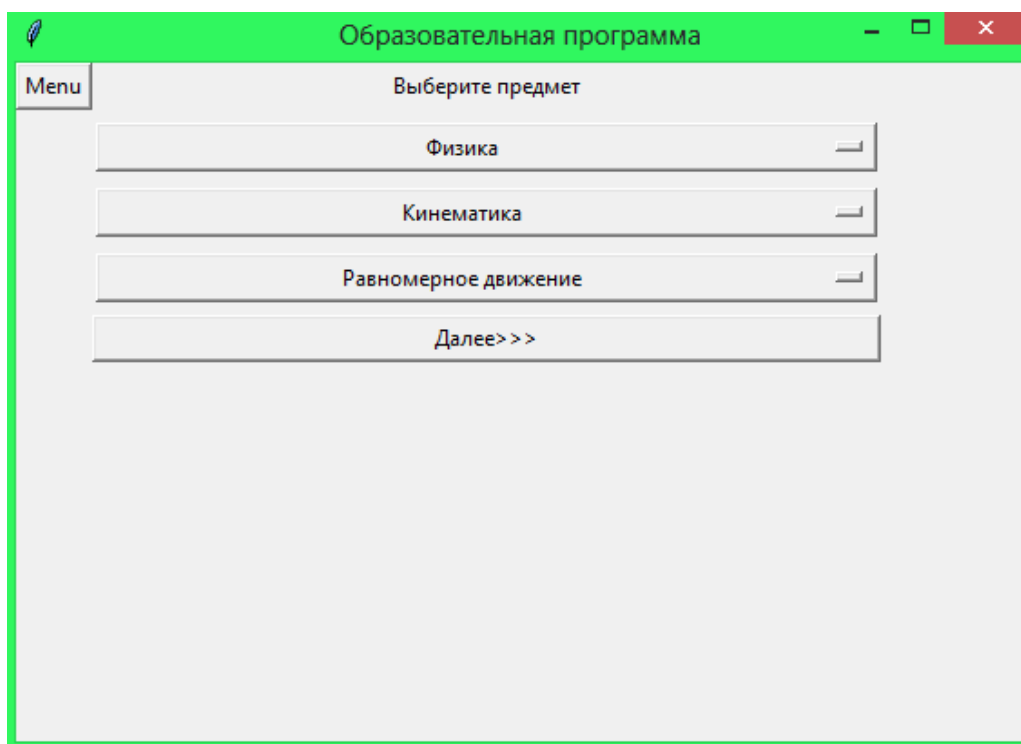


Рисунок №3 (окно выбора темы редактора)

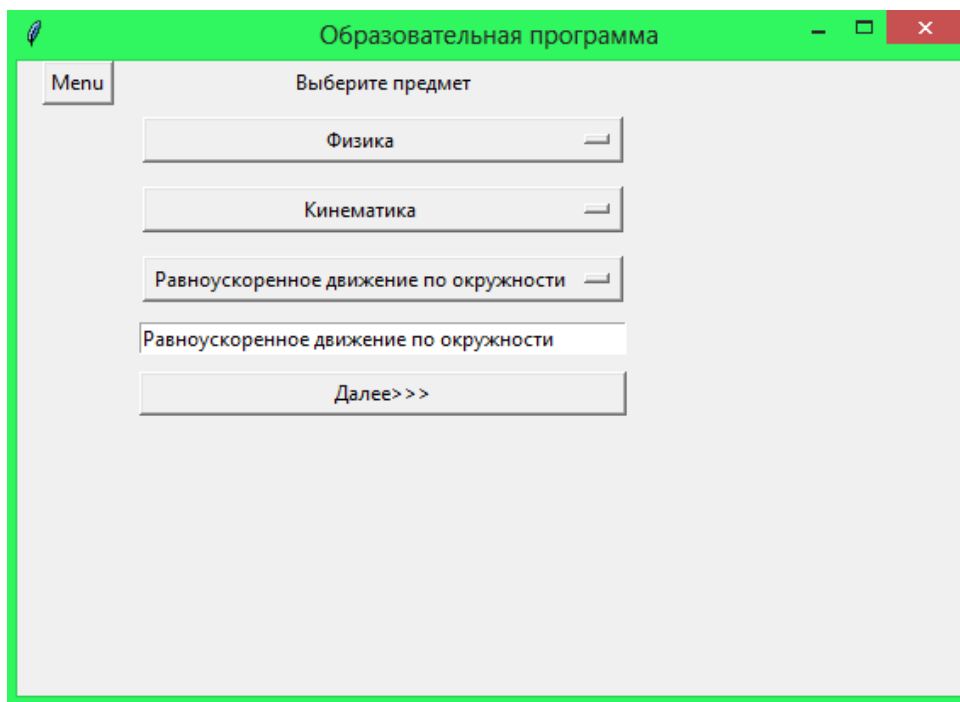


Рисунок №4 (редактор теоретического материала)

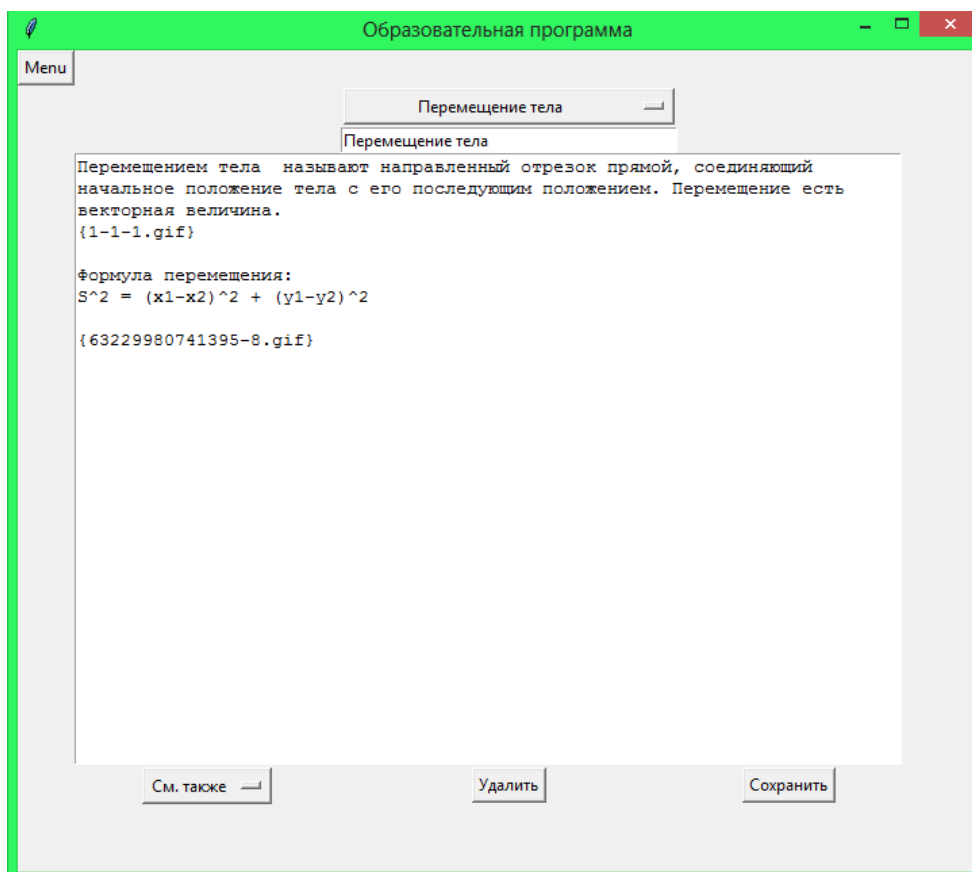


Рисунок №5 (окно просмотра теоретического материала)

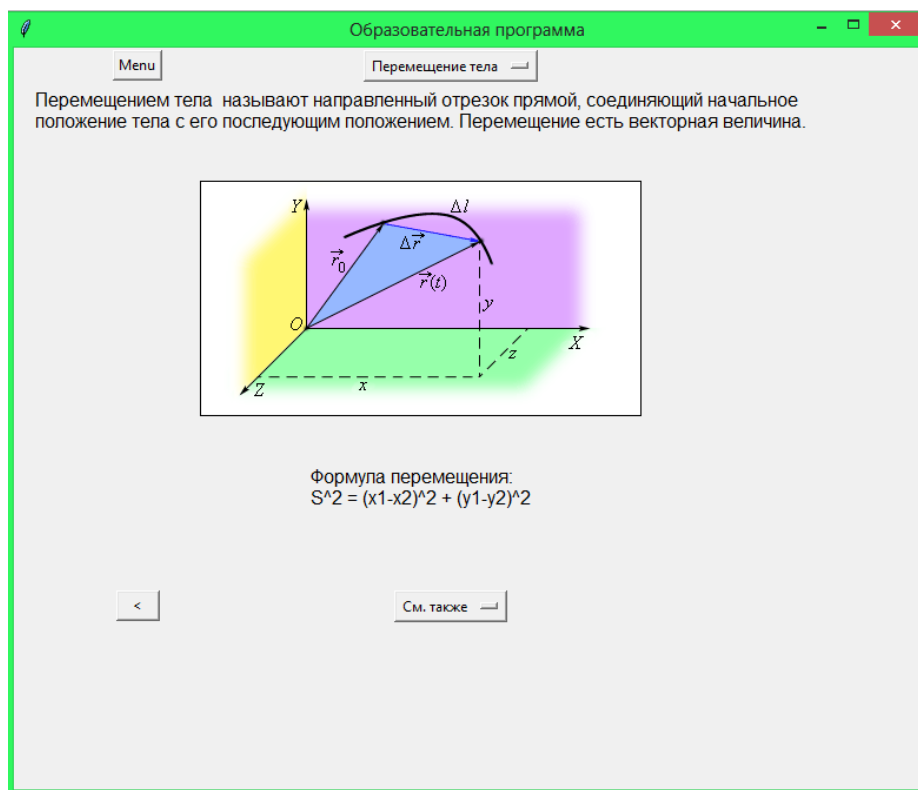


Рисунок №6 (редактор задач)

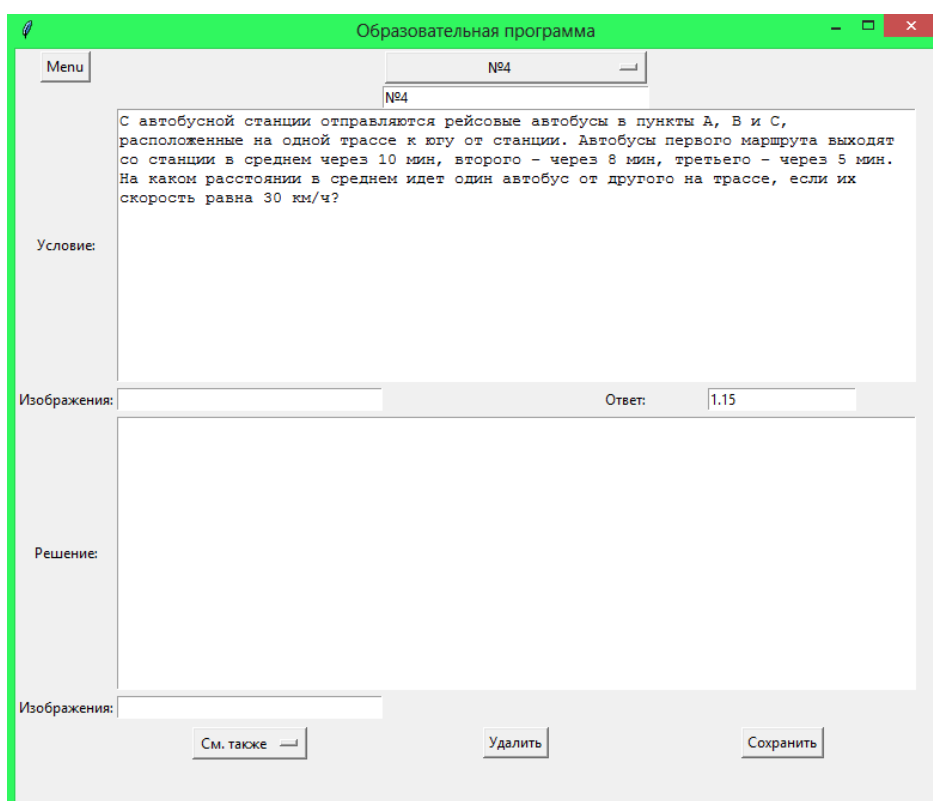


Рисунок №7 (окно решения задач)

The screenshot shows a window titled 'Образовательная программа' with a 'Menu' button in the top-left corner. Below the menu is a dropdown menu currently showing '№4'. The main text area contains the following problem: 'С автобусной станции отправляются рейсовые автобусы в пункты А, В и С, расположенные на одной трассе к югу от станции. Автобусы первого маршрута выходят со станции в среднем через 10 мин, второго – через 8 мин, третьего – через 5 мин. На каком расстоянии в среднем идет один автобус от другого на трассе, если их скорость равна 30 км/ч?'. Below the text is an input field labeled 'Ответ:' and a button labeled 'Ответить'. To the right of the 'Ответить' button is a small button with a right-pointing arrow '>'.

Рисунок №8 (редактор тестов)

The screenshot shows the same window 'Образовательная программа' but in a different state. The 'Menu' button is still present. Below it, there are two dropdown menus: the first shows '1' and the second shows 'A1'. The main text area contains a question: 'В течение какого времени скорый поезд длиной 150 м, идущий со скоростью 72 км/ч, будет проходить мимо товарного поезда длиной 300 м, идущего навстречу со скоростью 36 км/ч'. To the left of the text area is the label 'Вопрос:'. Below the text area is an input field labeled 'Изображения:'. Below this is a section titled 'Введите варианты ответа и отметьте правильный' (Enter answer options and mark the correct one). It contains four radio buttons with corresponding text: '10 с', '15 с', '30 с', and '20 д'. The '15 с' option is selected. At the bottom right is a button labeled 'Сохранить'.