



**Федеральное агентство по рыболовству**  
**Федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**

**«Астраханский государственный технический университет»**

Система менеджмента качества в области образования, воспитания, науки и инноваций сертифицирована DQS  
по международному стандарту ISO 9001:2015

Институт информационных технологий и коммуникаций  
Направление подготовки 09.03.04 Программная инженерия  
Профиль «Разработка программно-информационных систем»  
Кафедра «Автоматизированные системы обработки информации и управления»

## **КУРСОВОЙ ПРОЕКТ**

### **Программа-тренажер**

### **«Система двух уравнений с двумя неизвестными»**

по дисциплине «Программирование и информатика»

Допущен к защите  
«\_\_» \_\_\_\_\_ 20\_\_ г.  
Руководитель

Проект выполнен  
обучающимся группы ДИПРБ-11  
Иргалиевым А.А.

Оценка, полученная на защите  
«\_\_\_\_\_»

Руководитель  
ст. преп. Толасова В.В.

Члены комиссии:

\_\_\_\_\_ Лаптев В.В.  
\_\_\_\_\_ Куркурин Н.Д.  
\_\_\_\_\_ Толасова В.В.

- 1 Златопольский Д.М. Программирование: типовые задачи, алгоритмы, методы 2-е изд. – М.: БИНОМ. Лаборатория знаний, 2012. – 223 с.: ил.
- 2 Алгебра. 9 класс. Учеб. для учащихся общеобразоват. учреждений. В 2 ч. Ч. 2. [А.Г. Мордкович, П.В. Семёнов.] – 12-е изд. – М.: Мнемозина, 2010. – 224 с.: ил.

## УТВЕРЖДАЮ

Заведующий кафедрой

к.т.н., доцент

С.В. Белов \_\_\_\_\_

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

К заданию на курсовой проект  
по дисциплине  
«Программирование и информатика»

### КАЛЕНДАРНЫЙ ГРАФИК курсового проектирования

п/п	Разделы, темы и их содержание, графический материал	Дата сдачи	Объем, %
1	Выбор темы	05.10.2018	1
2	Техническое задание	12.10.2018	3
3	Разработка модели, проектирование системы <ul style="list-style-type: none"><li>▪ введение,</li><li>▪ технический проект,</li><li>▪ программа и методика испытаний,</li><li>▪ литература</li></ul>	09.11.2018	25
4	Программная реализация системы <ul style="list-style-type: none"><li>▪ работающая программа,</li><li>▪ рабочий проект</li><li>▪ скорректированное техническое задание (при необходимости)</li></ul>	30.11.2018	40
5	Тестирование и отладка системы, эксперименты <ul style="list-style-type: none"><li>▪ работающая программа с внесёнными изменениями,</li><li>▪ окончательные тексты всех разделов</li></ul>	14.12.2018	50
6	Компоновка текста Подготовка презентации и доклада <ul style="list-style-type: none"><li>▪ пояснительная записка</li><li>▪ презентация</li><li>▪ электронный носитель с текстом пояснительной записки, исходным кодом проекта, презентацией и готовым программным продуктом</li></ul>	18.12.2018	59
7	Защита курсового проекта	21.12.2018- 26.12.2018	60-100

С графиком ознакомлен « \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

Иргалиев А.А. \_\_\_\_\_, обучающийся группы ДИПРб-11  
(фамилия, инициалы, подпись)

График курсового проектирования выполнен  
без отклонений / с незначительными отклонениями / со значительными отклонениями  
нужное подчеркнуть

Руководитель курсового проекта \_\_\_\_\_ ст. преподаватель Толасова В.В.  
подпись, ученая степень, звание, фамилия, инициалы

## СОДЕРЖАНИЕ

Введение .....	5
1 Технический проект.....	6
1.1 Анализ предметной области .....	6
1.1.1 Система двух уравнений с двумя неизвестными .....	6
1.1.2 Тренировка и проверка знаний .....	7
1.2 Технология обработки информации .....	8
1.2.1 Форматы данных .....	8
1.2.2 Алгоритм вывода.....	9
1.2.3 Алгоритм генерации систем.....	10
1.2.4 Алгоритм вычисления числителя и знаменателя второго "y" .....	10
1.2.5 Алгоритм вычисления числителя и знаменателя второго "x" .....	10
1.2.6 Алгоритм тренировки .....	10
1.3 Входные и выходные данные .....	11
1.4 Системные требования.....	11
2 Рабочий проект.....	12
2.1 Общие сведения о работе системы .....	12
2.2 Функциональное назначение программного продукта.....	12
2.3 Инсталляция и выполнение программного продукта .....	12
2.4 Описание программы .....	13
2.5 Разработанные меню и интерфейсы .....	155
2.6 Сообщения системы .....	177
3 Программа и методика испытаний.....	18
3.1 Проверка работоспособности систем .....	18
3.2 Проверка работоспособности статистики .....	18
3.3 Проверка работоспособности справки .....	19
Заключение .....	20
Список использованных источников .....	21
Приложение 1 Техническое задание .....	22

## ВВЕДЕНИЕ

На итоговой аттестации в 9-х классах по модернизированным программам предлагаются задачи, в которых требуется решить системы алгебраических нелинейных уравнений. Школьники испытывают большие затруднения, встречаясь с такими заданиями. Этот раздел алгебры по праву считается одним из трудных, так как нет единых способов решения систем алгебраических уравнений. В наше время школьникам нужно постоянно практиковаться, решая подобные системы. Однако задачи в учебниках имеются в ограниченном количестве, и ребёнку не хватает примеров, чтобы разобраться с темой. Еще одна проблема учебников - это опечатки в ответах или заданиях. Ученик может решить систему уравнений правильно, но из-за опечатки в ответах он будет сомневаться в результатах своего решения.

Необходимо помочь школьникам преодолеть трудности при решении алгебраических систем нелинейных уравнений, научить отыскивать наиболее рациональный способ решения систем уравнений, тем самым подготовить выпускника основной школы к сдаче экзамена по математике, продолжению образования в выпускных классах средней школы с профильным обучением, а затем в вузе, где дисциплины математического цикла являются профильными.

Разрабатываемая в качестве курсового проекта программа-тренажёр предоставляет огромное множество задач для пользователя, что, несомненно, облегчит работу учителя, так как ребёнок будет сам обучаться и будет уверен в правильности постановки задания и его ответе.

Целью создания программы-тренажера «Система двух уравнений с двумя неизвестными» является автоматизация обучения решению данных систем. Назначение программы – повышение успеваемости и качества знаний школьников, снижение нагрузки на преподавателя.

Таким образом, разработка компьютерного тренажёра, предлагающего задание и проверяющего правильность его выполнения позволит школьнику подготовиться к зачёту самостоятельно.

## 1 ТЕХНИЧЕСКИЙ ПРОЕКТ

### 1.1 Анализ предметной области

#### 1.1.1 Система двух уравнений с двумя неизвестными

Уравнение — это равенство, содержащее неизвестное число, которое надо найти. Неизвестные числа в уравнениях принято обозначать **x**, **y**, **z**. Решение уравнения — это задача по нахождению таких значений аргументов, при которых выполняется равенство (выражения слева и справа от знака равенства становятся эквивалентными). Значения неизвестных переменных, при которых это равенство достигается, называются решениями или корнями данного уравнения. Система уравнений – это два и более уравнения, которыми можно манипулировать для нахождения общих решений. Система из двух уравнений включает в себя две переменные, значения которых являются общими для обоих уравнений.

$$\begin{cases} ax + bxy + y^2 = 0, \\ dx - fy = c \end{cases} \quad (1)$$

где

**x**, **y** – переменные,

**a**, **b**, **c**, **d**, **f** – коэффициенты.

Решением системы уравнений с двумя неизвестными называется пара значений этих неизвестных, обращающая каждое уравнение системы в верное числовое равенство, другими словами, являющаяся решением каждого уравнения системы.

Такая система может иметь до двух решений, так как одно из уравнений квадратное. В данном же случае, решением системы являются две точки (**x**, **y**) и (**x**<sub>2</sub>, **y**<sub>2</sub>). Координаты первой точки всегда целые, а координаты второй – чаще всего дробные. Решить такую систему можно разными методами: графическим, методом ввода новых переменных, алгебраического сложения, подстановки и так далее.

Системы двух уравнений, одно из которых квадратное, а другое – линейное (система 1), как правило, решаются методом подстановки одного из неизвестных (полученного из линейного уравнения) в квадратное уравнение, и задача сводится к решению квадратного уравнения с одним неизвестным.

Предполагается, что программа-тренажер будет генерировать координаты первой точки (**x**, **y**) из целочисленного диапазона  $x \in [-4, 0) \cup (0, 4]$  и  $y \in [-5, 0) \cup (0, 5]$  и коэффициенты **a**, **b** из целочисленного диапазона  $[-10, 0) \cup (0, 10]$  до тех пор, пока квадратное уравнение системы (1) не будет выполняться. После генерировать коэффициенты **d** и **f** из того же диапазона, что и коэффициенты первого уравнения и вычислять коэффициент **c** по линейному уравнению системы (1).

Известно, что данная система имеет два решения (одно из которых дробное). Целые координаты первой точки будет генерировать программа, а для того, чтобы найти дробные координаты второй точки обратимся к теореме Виета.

**Теорема Виета:** Сумма корней приведенного квадратного трёхчлена  $x^2 + px + q = 0$  равна его второму коэффициенту  $p$  с противоположным знаком, а произведение – свободному члену  $q$ . В случае неприведенного квадратного уравнения  $ax^2 + bx + c = 0$  формулы Виета имеют вид:

$$x_1 + x_2 = -\frac{b}{a}, \quad (2)$$

где

$b, a$  – коэффициенты,

$x_1, x_2$  – абсциссы двух точек.

$$x_1 x_2 = \frac{c}{a}, \quad (3)$$

где

$c, a$  – коэффициенты,

$x_1, x_2$  – абсциссы двух точек.

Так как нам нужно найти координаты второй точки вычисляем по формуле 3 координату абсциссы:

$$x_2 = \frac{c}{ax_1}, \quad (4)$$

где

$c, a$  – коэффициенты,

$x_2, x_1$  – абсциссы двух точек.

А для того, чтобы найти ординату второй точки подставляем формулу 4 в линейное уравнение системы 1 и вычисляем  $y_2$ :

$$y_2 = \frac{c(d - ax_1)}{ax_1 f}, \quad (5)$$

где

$c, d, a, f$  – коэффициенты,

$x_1, y_2$  – переменные.

### 1.1.2 Тренировка и проверка знаний

Приступая к тренировке, обучающемуся необходимо повторить теоретический материал. Он должен знать определенные формулы и методы решения и уметь их применять на практике. Без них учащийся не сможет справиться с данным заданием.

Количество заданий не должно быть слишком велико, чтобы не вызвать утомления, однако и не должно быть слишком мало. Нужно быть уверенным, что материал действительно

усвоен пользователем. Данная система должна состоять из двух уравнений, одно из которых линейное, а второе - квадратное. Коэффициенты при неизвестных должны быть в диапазоне от -10 до 10, чтобы не вызывать у учащегося трудностей во время вычислений.

Подсчёт подряд идущих правильных или неправильных ответов лучше всего покажет уровень знаний обучающегося в данной области. Если серия правильных ответов имеет длину 5, то можно сказать, что данный материал достаточно усвоен. Если серия неправильных ответов имеет длину 3, то рекомендуется повторить теорию и приступить к тестированию снова. Если серии имеют длину меньше указанной, то следует продолжать тренировку. По желанию обучающегося предусмотрено прекращение тренировки.

В случае правильного ответа желательно сообщать об этом пользователю, а в случае неверного ответа – выдавать правильный ответ.

## 1.2 Технология обработки информации

Анализ предметной области показал, что программа рассчитана на одного пользователя.

На рис. 1.1. показана диаграмма вариантов использования.

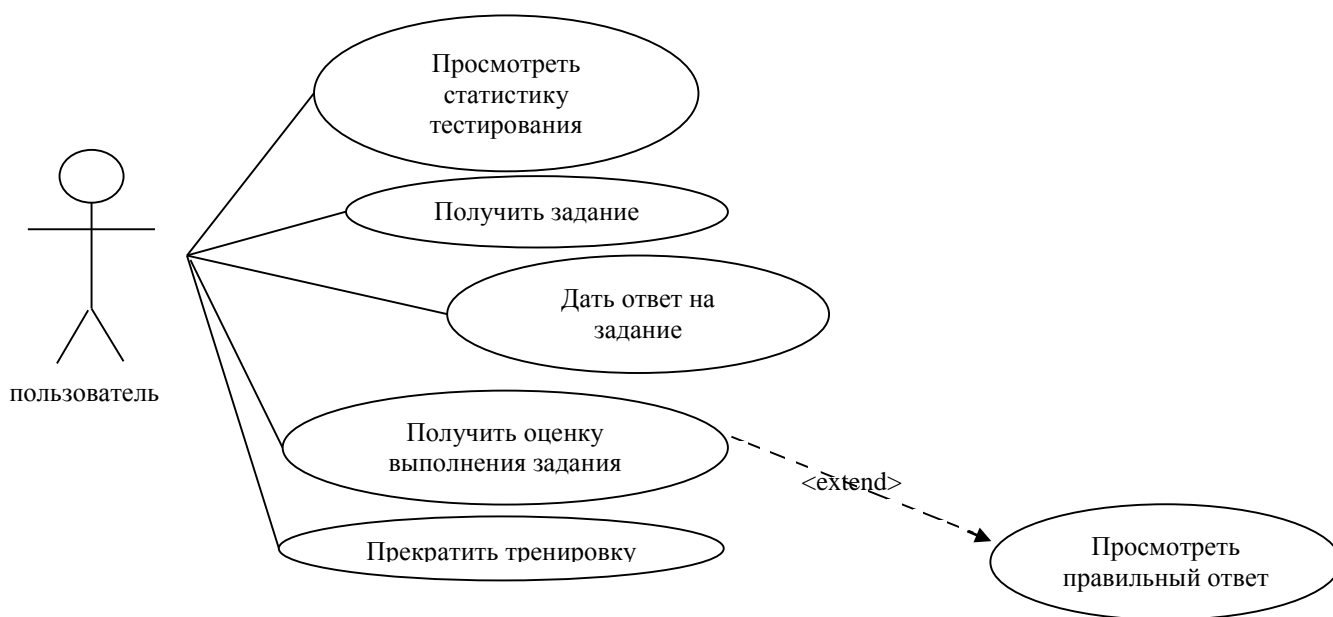


Рисунок 1.1 – Диаграмма вариантов использования

Вариант использования «Получить оценку выполнения задания» расширен функцией «Просмотреть правильный ответ»

### 1.2.1 Форматы данных

Задание представляет собой систему, содержащую два уравнения, одно из которых квадратное, другое – линейное, при этом коэффициенты и точки  $(x, y)$  и  $(x_2, y_2)$  не равны 0. Данную систему нужно решить, найдя точки  $(x, y)$  и  $(x_2, y_2)$  (если она есть).

Ниже приведён пример задания.



Решите следующую систему:

$$8x - 2xy + y * y = 0$$

$$-2x - 3y = 14$$

### 1.2.2 Алгоритм вывода

#### Уравнение №1

1. Проверка коэффициента **a** на знак.
  - Если коэффициент отрицательный, то вывести перед ним знак «-».
  - Иначе вывести **a**.
2. Если абсолютное значение коэффициента равно 1, то не выводить.
3. Вывести «x».
4. Проверка коэффициента **b** на знак.
  - Если коэффициент отрицательный, а в уравнении перед ним стоит «+», то вывести перед **b** знак «-».
  - Иначе вывести **b**.
5. Если абсолютное значение коэффициента равно 1, то не выводить.
6. Вывести « $xу + y * y = 0$ ».
7. Конец алгоритма.

#### Уравнение №2

1. Проверка коэффициента **d** на знак.
  - Если коэффициент отрицательный, то вывести перед ним знак «-».
  - Иначе вывести **d**.
2. Если абсолютное значение коэффициента равно 1, то не выводить.
3. Вывести «x».
4. Проверка коэффициента **f** на знак.
  - Если коэффициент отрицательный, а в уравнении перед ним стоит «-», то вывести перед **f** знак «+».
  - Иначе вывести **f**.
5. Если абсолютное значение коэффициента равно 1, то не выводить.
6. Вывести « $y =$ ».
7. Вывести коэффициент **c**.
8. Конец алгоритма.

**1.2.3 Алгоритм генерации систем**

1. Сгенерировать два случайных целых числа  $a$  и  $b$  из диапазона  $[-10,0) \cup (0,10]$  и точку  $(x,y)$  из диапазона  $x \in [-4,0) \cup (0,4]$  и  $y \in [-5,0) \cup (0,5]$ .
2. Если  $ax + bxy + y^2 = 0$ , то перейти к п. 3, иначе – перейти к п. 1.
3. Сгенерировать два случайных целых числа  $d$  и  $f$  из диапазона  $[-10,0) \cup (0,10]$ .
4. Присвоить  $c = dx - fy$ .
5. Вывести оба уравнения, то есть систему двух уравнений на экран по алгоритму **1.2.2**.
6. Конец алгоритма.

**1.2.4 Алгоритм вычисления числителя и знаменателя второго «у»**

1. Присвоить  $NumY2 = a * c$ .
2. Присвоить  $DenomY2 = n * y$ , где  $n = b * f + d$ .
3. Если знаменатель отрицательный, то поменять знаки у числителя и знаменателя на противоположные.

**1.2.5 Алгоритм вычисления числителя и знаменателя второго «х»**

1. Присвоить  $NumX2 = c * DenomY2 + f * NumY2$ .
2. Присвоить  $DenomX2 = d * DenomY2$ .
3. Если знаменатель отрицательный, то поменять знаки у числителя и знаменателя на противоположные.

**1.2.6 Алгоритм тренировки**

Запросить у пользователя, хочет ли он открыть справку или посмотреть статистику.

Запросить у пользователя, начать ли тестирование.

Счётчик серии правильных ответов  $P$  изначально равен нулю.

Счётчик серии неправильных ответов  $N$  также равен нулю.

Переменная  $E := 1$ .

Пока  $E = 1$  и  $P$  меньше 5 и  $N$  меньше 3, повторять

- | Вызвать вспомогательный алгоритм генерации задания **1.2.3**.
- | Вызвать вспомогательный алгоритм вычисления второго «у» **1.2.4**.
- | Сократить числитель и знаменатель на их НОД, который вычислить по алгоритму Евклида.
- | Вызвать вспомогательный алгоритм вычисления второго «х» **1.2.5**.
- | Сократить числитель и знаменатель на их НОД, который вычислить по алгоритму Евклида.
- | Запросить у пользователя корни уравнений, которые он получил.
- | Если задание выполнено верно, то
  - | | увеличить  $P$  на 1,
  - | | присвоить  $N$  значение 0,

```
|   иначе
|   |   Вывести правильный ответ,
|   |   увеличить  $N$  на 1,
|   |   присвоить  $P$  значение 0,
|   конец ветвления
|   Запросить у пользователя, хочет ли он продолжать работу (величина  $E$  должна быть
|   равна единице, если пользователь хочет продолжать тренировку и любому другому
|   целому числу, если пользователь хочет завершить тренировку и выйти в меню)
конец цикла
Если  $E \neq 1$  то
|   сообщить пользователю о завершении тренировки по его желанию
А если  $P = 5$ 
|   сообщить пользователю об успешном окончании тренировки
А если  $N = 3$ 
|   сообщить пользователю, что ему нужно отдохнуть, повторить методы решения
|   систем
|   двух уравнений с двумя неизвестными и пройти тренировку заново.
конец ветвления
```

### 1.3 Входные и выходные данные

Входные данные:

- выбор пунктов меню;
- решение системы, полученное пользователем;
- ответ пользователя на вопрос, продолжать ли работу с тренажёром;
- ответ пользователя на вопрос, координаты второй точки целые или дробные.

Выходные данные:

- система двух уравнений с двумя неизвестными;
- ответ на задание;
- результат тренировки.

### 1.4 Системные требования

Рекомендуемая конфигурация:

- Intel-совместимый процессор с частотой не менее 1,6 ГГц;
- не менее 512 МБ ОЗУ;
- не менее 20 МБ свободного места на диске;
- дисковод CD-ROM/DVD-ROM.

Операционная система: Windows XP (x86) с пакетом обновления 3 (SP3). Среда разработки – Semantic IDE.

## **2 РАБОЧИЙ ПРОЕКТ**

### **2.1 Общие сведения о работе системы**

Программный продукт разработан в интегрированной среде Semantic IDE (версия 1.0.3.2) на языке Semantic Language. Программа работает под управлением операционной системы Windows XP (x86) Professional (SP3) и более поздними. Стартовый модуль для Semantic IDE – kurs.sl.

### **2.2 Функциональное назначение программного продукта**

Разработанный программный продукт предназначен для отработки навыков решения систем двух уравнений с двумя неизвестными. Программа имеет следующие функциональные возможности:

- предоставление пользователю статистики;
- проверка правильности вычисления пользователем предложенного задания;
- вывод правильного ответа в случае ошибки;
- прекращение тренировки:
  - по желанию пользователя (результат не оценивается)
  - при достаточном количестве подряд идущих верных ответов;
  - в случае серии неправильных ответов.

### **2.3 Установка и выполнение программного продукта**

Для функционирования программы на компьютере должен быть установлен MS .NET Framework 4.0, обеспечивающий работу интегрированной среды Semantic IDE.

Для выполнения программы необходимо:

1. Скопировать на жесткий диск компьютера папку sys.eq, содержащую проект Soe.prj, модуль kurs.sl и текстовый документ savest для статистики.
2. Запустить интегрированную среду Semantic IDE
3. Открыть из среды файл Soe.prj.
4. Убедиться, что модуль kurs.sl назначен стартовым.
5. Запустить на выполнение модуль kurs.sl.

## 2.4 Описание программы

В таблице 2.1 приведены функции и процедуры, используемые в программе.

Таблица 2.1 – Функции и процедуры программы

Прототип	Назначение
закр функция (входной целое d): целое unzero	Проверка на ненулевые значения коэффициентов и корней уравнения
закр процедура (входной целое a, входной целое b): Equation1	Вывод на экран квадратного уравнения системы
закр процедура (входной целое f, входной целое d, входной целое c): Equation2	Вывод на экран линейного уравнения системы
закр процедура (переменный целое x, переменный целое y, переменный целое a, переменный целое b, переменный целое c, переменный целое d, переменный целое f): Generation	Генерирует коэффициенты (a, b, c, d, f) и точку (x,y)
закр функция (входной целое Num, входной целое Denom) целое NOD	Вычисляет Наибольший Общий Делитель числителя и знаменателя (для сокращения дроби)
закр процедура (входной целое a, входной целое c, входной целое n, входной целое y, переменный целое NumY2, переменный целое DenomY2) SecondY	Вычисляет числитель и знаменатель ординаты второй точки
закр процедура (входной целое P, входной целое N) SaveStatic	Сохранение статистики тренировок в текстовый документ
закр процедура (переменный булев menu) Main	Подсчёт серий правильных и неправильных ответов, в зависимости от ответа пользователя. При том или ином исходе вывод на экран сообщений программы
закр процедура () DisplayStatic	Вывод статистики на экран

Продолжение таблицы 2.1

Прототип	Назначение
закр процедура (входной целое с, входной целое d, входной целое f, входной целое DenomY2, входной целое NumY2, переменный целое NumX2, переменный целое DenomX2) SecondX	Вычисляет числитель и знаменатель абсциссы второй точки
закр процедура (входной целое Num, входной целое Denom) ReadFraction	Считывает числитель и знаменатель с клавиатуры
закр функция (входной целое x, входной целое y, входной целое NumX2, входной целое DenomX2, входной целое NumY2, входной целое DenomY2) булев Check	Проверяет ответ пользователя с эталонным

В таблице 2.2 приведены важнейшие переменные, используемые в программе.

Таблица 2.2 – Важнейшие переменные программы

Имя	Тип	Назначение
P	целое	Длина серии верных ответов
N	целое	Длина серии ошибочных ответов
E	целое	Ответ пользователя на вопрос задания
x	целое	Правильный ответ абсциссы первой точки
y	целое	Правильный ответ ординаты первой точки
NumX2	целое	Числитель абсциссы второй точки
DenomX2	целое	Знаменатель абсциссы второй точки
NumY2	целое	Числитель ординаты второй точки
DenomY2	целое	Знаменатель ординаты второй точки
x2	целое	Правильный ответ абсциссы второй точки
y2	целое	Правильный ответ ординаты второй точки

## 2.5 Разработанные меню и интерфейсы

После запуска программы на выполнение в консольном окне появится меню (рис.2.1), которое позволяет выбрать один из трёх видов упражнений либо выйти из программы.

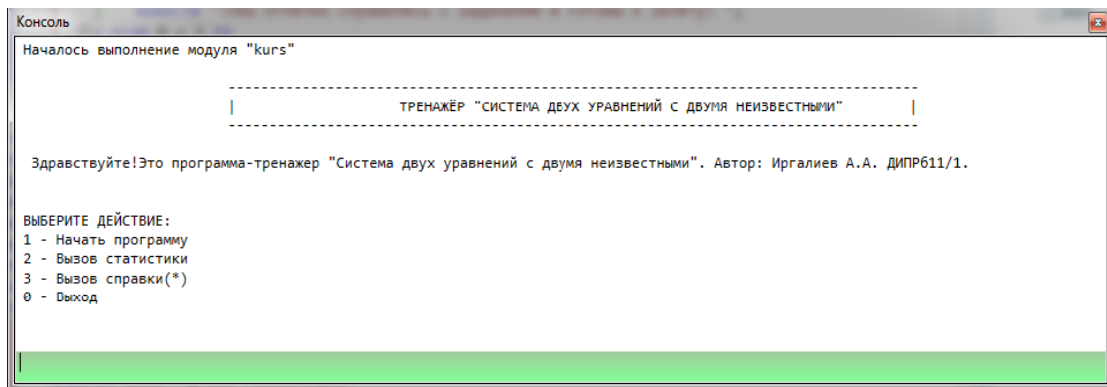


Рисунок 2.1 – Консоль программы с выбором задания

При выборе пункта 2 пользователю будет показана статистика тренировок (рис. 2.2).

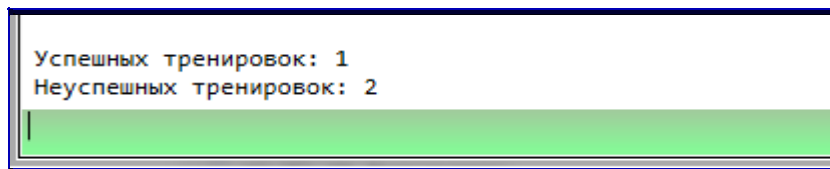


Рисунок 2.2 – Консоль программы со статистикой пользователя

При выборе пункта 3 пользователю будет показана справка по программе (рис.2.3).

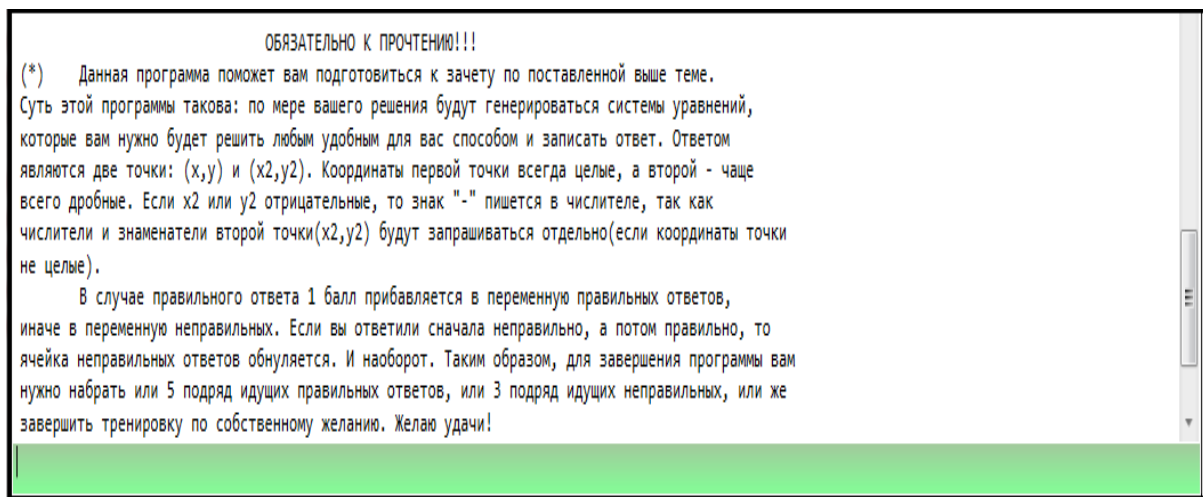


Рисунок 2.3 – Консоль программы со справкой

При выборе пункта 1 пользователю будут выдаваться системы двух уравнений с двумя неизвестными (начало тренировки)(рис.2.4).

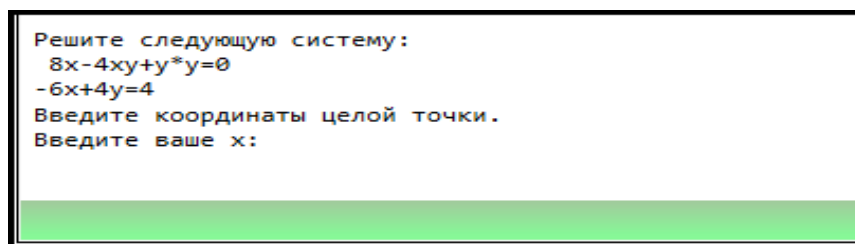


Рисунок 2.4 – Консоль программы с заданием

При вводе пользователем результатов решения системы, программа запрашивает координаты второй точки: целые или дробные (рис. 2.5).

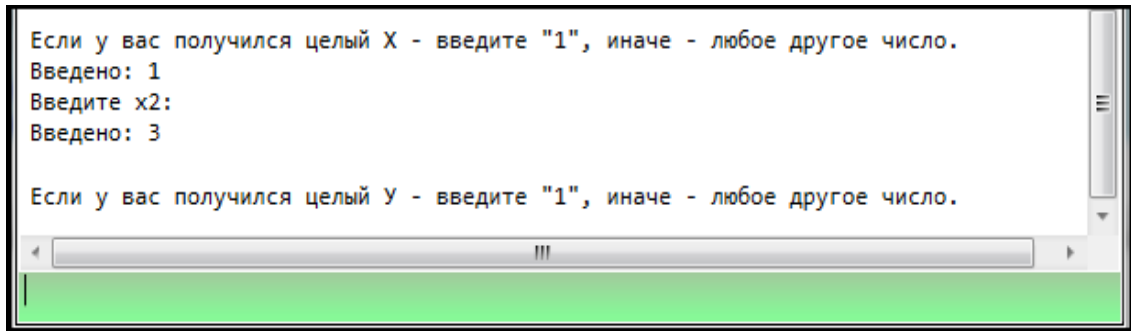


Рисунок 2.5 – Консоль программы с запросом координат второй точки.

После того, как пользователь введёт ответ, ему будут сообщены результаты проверки. Если пользователь правильно ответил на вопрос, а серия правильных ответов не достигла пяти, ему будет предложено продолжение тренировки и очередное задание. Если пользователь ошибся, будет выведен верный ответ и запрошено подтверждение о продолжении тренировки, показанное на рисунке 2.6.

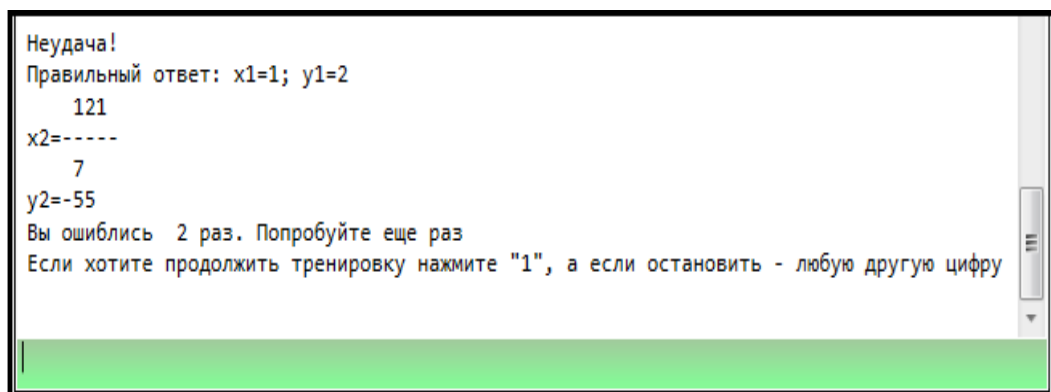


Рисунок 2.6 – Запрос о продолжении тренировки

Тренировка будет продолжаться до тех пор, пока не будет введено любое число, кроме 1 на вопрос о продолжении, либо не будут даны три неправильных ответа подряд, либо не будут даны подряд пять верных ответов (рис. 2.7).

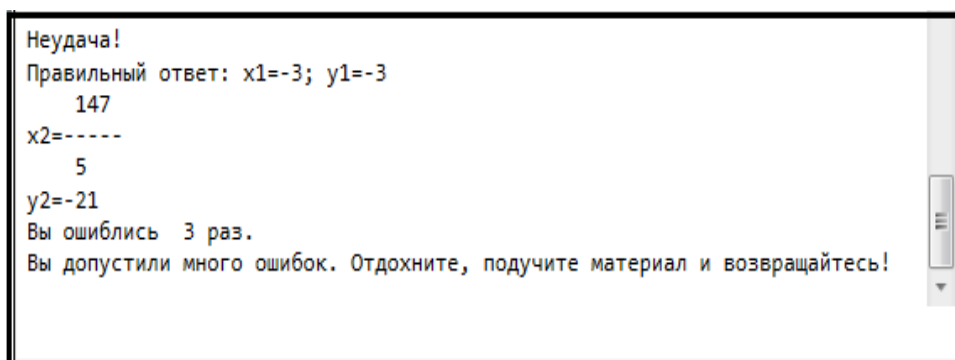


Рисунок 2.7 – Окончание тренировки в результате серии неправильных ответов



## 2.6 Сообщения системы

В таблице 2.3 приведены сообщения системы.

Таблица 2.3 – Сообщения системы

№ п\п	Сообщение	Причина возникновения, способ устранения
1	Правильно!	Пользователь ответил верно
2	Неудача! Правильный ответ: $x=...$ $y=...$ $x^2=...$ $y^2=...$ Попробуйте еще раз	Пользователь допустил ошибку
3	Программа была остановлена по собственному желанию пользователя	Пользователь захотел прекратить программу в целом
4	Вы допустили много ошибок. Отдохните, подучите материал и возвращайтесь!	Тренировка прекращена из-за большого количества ошибок
5	Вы отлично справились с заданиями и готовы к зачёту!	Тренировка прекращена по причине уверенных знаний пользователя
6	Вы закончили тренировку по собственному желанию. Обязательно возвращайтесь!	Пользователь захотел прекратить тренировку

В случае появления других сообщений следует обратиться к разработчику.

### 3 ПРОГРАММА И МЕТОДИКА ИСПЫТАНИЙ

#### 3.1 Проверка работоспособности систем

1. Запустить программу на выполнение. Появится меню (см. рис. 2.1).
2. Выбрать пункт 1, убедиться, что предложено задание системы (см. рис. 2.4)
3. Ввести правильный ответ, убедиться, что получено сообщение 1 (см. табл. 2.3), а также запрос о продолжении или прекращении тренировки (см. рис. 2.5).
4. Ввести единицу и убедиться, что программа продолжила работу и выдано новое задание.
5. Повторить пункты 3 и 4 ещё 4 раза, убедиться, что программа выдала сообщение 5 (см. табл. 2.3) и завершила работу.
6. Запустить программу на выполнение, выбрать пункт 1.
7. После получения задания ввести неправильный ответ, убедиться, что получено сообщение 2 (см. табл. 2.3), а также запрос о продолжении или прекращении тренировки (см. рис. 2.5).
8. Повторить пункт 4 и убедиться, что выдано новое задание (см. рис. 2.4).
9. Повторить пункты 8 и 9 ещё 2 раза, убедиться, что программа выдала сообщение 4 (см. табл. 2.3) и завершила работу (см. рис. 2.6).
10. Запустить программу на выполнение, выбрать пункт 1. При получении заданий чередовать правильные и неправильные ответы так, чтобы длина серии верных ответов не превышала 4, а длина серии неверных ответов не превышала 2. Убедиться, что выход из программы не происходит (при условии, что на запрос о продолжении введено число «1»).
11. При запросе на продолжение ввести число, отличное от единицы – убедиться, что программа завершила тренировку с сообщением 6 (см. табл. 2.3) и вышла в меню (см. рис. 2.1).
12. По желанию, в меню выбрать пункт 2, для показа статистики тренировок (см. рис. 2.2).
13. Нажать «0» для завершения программы, убедиться, что получено сообщение 3 (см. табл. 2.3).

#### 3.2 Проверка работоспособности статистики

1. Запустить программу на выполнение. Появится меню (см. рис. 2.1).
2. Выбрать пункт 2, убедиться, что показана статистика тренировок (см. рис. 2.2).
3. Выбрать пункт 1, убедиться, что предложено задание системы (см. рис. 2.4).

4. Ввести правильный ответ, убедиться, что получено сообщение 1 (см. табл. 2.3), а также запрос о продолжении или прекращении тренировки (см. рис. 2.5).
5. Ввести единицу и убедиться, что программа продолжила работу и выдано новое задание.
6. Повторить пункты 4 и 5 еще 4 раза, убедиться, что программа выдала сообщение 5 (см. табл. 2.3) и завершила работу.
7. Запустить программу на выполнение, выбрать пункт 2.
8. Убедиться, что количество успешных тренировок увеличилось на 1.
9. Выбрать пункт 1, убедиться, что предложено задание системы (см. рис. 2.4).
10. Ввести неправильный ответ, убедиться, что получено сообщение 2 (см. табл. 2.3), а также запрос о продолжении или прекращении тренировки (см. рис. 2.5).
11. Ввести единицу и убедиться, что программа продолжила работу и выдано новое задание.
12. Повторить пункты 10 и 11 еще 2 раза, убедиться, что программа выдала сообщение 4 (см. табл. 2.3) и завершила работу.
13. Запустить программу на выполнение. Появится меню (см. рис. 2.1).
14. Выбрать пункт 2, убедиться, что количество неуспешных тренировок увеличилось на 1.
15. Нажать «0» для завершения программы в целом, убедиться, что получено сообщение 3 (см. табл. 2.3).

### **3.3 Проверка работоспособности справки**

1. Запустить программу на выполнение. Появится меню (см. рис. 2.1).
2. Выбрать пункт 3, убедиться, что предложена справка (см. рис. 2.3).
3. Убедиться, что всё написано доходчиво и приступить к тестированию (см. подпункт 3.1).
4. Нажать «0» для завершения программы и убедиться, что получено сообщение 3 (см. табл. 2.3).

### **ЗАКЛЮЧЕНИЕ**

В результате курсового проектирования разработана программа-тренажёр «Система двух уравнений с двумя неизвестными». Тренажёр предлагает задания на знание методов решения такой системы и проверяет правильность ответа, в случае ошибки сообщает правильный ответ. Тренировка продолжается до получения серии из пяти правильных либо трёх неправильных ответов, по окончании выдаётся сообщение о том, насколько успешно были выполнены задания.

Программа отвечает поставленным требованиям и может быть использована для обучения школьников 7-9 классов.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Златопольский Д.М. Программирование: типовые задачи, алгоритмы, методы 2-е изд. – М.: БИНОМ. Лаборатория знаний, 2012. – 223 с.: ил.
2. Алгебра. 9 класс. Учеб. для учащихся общеобразоват. учреждений. В 2 ч. Ч. 2. [А.Г. Мордкович, П.В. Семёнов.] – 12-е изд. – М.: Мнемозина, 2010. – 224 с.: ил.
3. Мазярчук Л.В. Подготовка к ГИА. Занятие мобильной группы по теме "Решение систем уравнений с двумя переменными"– [Электронный ресурс] режим доступа: <http://открытыйурок.рф/статьи/589342/> (11.01.2011)

**ПРИЛОЖЕНИЕ 1****ТЕХНИЧЕСКОЕ ЗАДАНИЕ****на выполнение курсового проекта**

по дисциплине «Программирование и информатика»

Направление 090304 – Программная инженерия

Исполнитель: обучающийся гр. ДИПРб11 Иргалиев А.А.

Тема: **Программа-тренажёр «Система двух уравнений с двумя неизвестными»****1 Назначение, цели и задачи разработки**

**Цель разработки:** автоматизация обучения решению системы двух уравнений с двумя неизвестными.

**Назначение разработки:**

- повышение качества знаний пользователей;
- снижение нагрузки на преподавателя.

**Основные задачи,** решаемые разработчиком в процессе выполнения курсового проекта:

- анализ предметной области;
- разработка программного продукта в соответствии с требованиями;
- документирование проекта в соответствии с установленными требованиями.

**2 Характер разработки:** прикладная квалификационная работа.**3 Основания для разработки**

- Учебный план направления 09.03.04 «Программная инженерия» 2018 года набора.
- Рабочая программа дисциплины Программирование и информатика».
- Распоряжение по кафедре АСОИУ №\_\_\_\_ от «\_\_»\_\_\_\_\_ 2018 г.

**4 Плановые сроки выполнения** – осенний семестр 2017/18 учебного года:

Начало «05» октября 2018 г.

Окончание «18» декабря 2018 г.

**5 Требования к проектируемой системе****5.1 Требования к функциональным характеристикам**

Проектируемая система должна обеспечивать выполнение следующих основных функций:

- предоставлять пользователю системы двух уравнений с двумя неизвестными вида:

$$\begin{cases} ax + by + y^2 = 0 \\ dx - fy = c \end{cases}$$

- запрашивать у пользователя ответ на предложенное задание, то есть «х» и «у»;
- проверять правильность ответа и выдавать правильный ответ в случае ошибки;
- Предоставлять пользователю статистику его тренировок.

- прекращать тренировку в одном из случаев:
  - пользователь безошибочно выполнил 5 заданий подряд (успешная тренировка);
  - пользователь ошибся 3 раза подряд (неуспешная тренировка);
  - пользователь решил прекратить занятие (тренировка прервана, результат не оценивается).

Система предусматривает функциональные ограничения:

- коэффициенты уравнений  $a$ ,  $b$ ,  $d$ ,  $f$  целые и расположены в диапазоне  $[-10; 0) \cup (0; 10]$ ;
- программа не может использоваться в качестве автоматического нахождения корней уравнений.

## 5.2 Требования к эксплуатационным характеристикам

Программа не должна аварийно завершаться при любых действиях пользователя

Время реакции программы на действия пользователя не должно превышать 10 секунд.

## 5.3 Требования к программному обеспечению:

Среда разработки – Semantic IDE

Операционная система: Windows XP (x86) с пакетом обновления 3 (SP3)

## 5.4 Требования к аппаратному обеспечению:

Рекомендуемая конфигурация:

- Intel-совместимый процессор с частотой не менее 1,6 ГГц;
- не менее 512 МБ ОЗУ;
- не менее 20 МБ свободного места на диске;
- дисковод CD-ROM/DVD-ROM.

# 6 Стадии и этапы разработки

## 6.1 Эскизный проект (ЭП)

- Анализ предметной области.
- Подготовка проектной документации.

## 6.2 Технический проект (ТП)

- Разработка структур и форм представления данных.
- Разработка структуры программного комплекса.
- Подготовка пояснительной записки.

## 6.3 Рабочий проект (РП)

- Программная реализация.
- Тестирование и отладка программы.
- Подготовка программной и эксплуатационной документации.

## **6.4 Эксплуатация (Э)**

Описание и анализ результатов проведенного исследования.

## **7 Требования к документированию проекта**

К защите курсового проекта должны быть представлены следующие документы:

- Пояснительная записка к курсовому проекту;
- Презентация доклада.
- Программа, презентация и пояснительная записка к курсовому проекту на оптическом носителе.

Требования к структуре документов определены соответствующими стандартами ЕСПД.

Требования к оформлению определены соответствующими методическими указаниями.

## **8 Порядок контроля и приемки**

Контроль выполнения курсового проекта проводится руководителем поэтапно в соответствии с утвержденным графиком выполнения проекта.

На завершающем этапе руководитель осуществляет нормоконтроль представленной исполнителем документации и принимает решение о допуске (недопуске) проекта к защите.

Защита курсового проекта проводится комиссией в составе не менее двух человек, включая руководителя проекта.

В процессе защиты проекта исполнитель представляет документацию, делает краткое сообщение по теме разработки и демонстрирует ее программную реализацию.

При выставлении оценки учитывается:

- степень соответствия представленной разработки требованиям технического задания;
- качество программной реализации, документации и доклада по теме проекта;
- соблюдение исполнителем графика выполнения курсового проекта.

## **9 Литература**

1. Златопольский Д.М. Программирование: типовые задачи, алгоритмы, методы 2-е изд. – М.: БИНОМ. Лаборатория знаний, 2012. – 223 с.: ил.
2. Алгебра. 9 класс. Учеб. для учащихся общеобразоват. учреждений. В 2 ч. Ч. 2. [А.Г. Мордкович, П.В. Семёнов.] – 12-е изд. – М.: Мнемозина, 2010. – 224 с.: ил.