Содержание

Введение 7

1 Характеристика места практики 9

2 Сведения об имеющейся вычислительной технике и программном обеспечении 11

3 Сведения о прохождении практики в качестве дублера инженерно-технического работника 14

4 Индивидуальное задание 15

4.1 Постановка задачи 15

4.2 Проектирование структуры программы 16

4.2.1 Desktop версия приложения 16

4.2.2 Мобильная версия приложения 16

4.3 Функциональное взаимодействие модулей программы 16

4.3.1 Desktop версия приложения 16

4.3.2 Мобильная версия приложения 18

4.4 Разработка unit-тестов 21

4.5 Работа с программой 22

4.5.1 Desktop версия приложения 22

4.5.2 Мобильная версия приложения 25

Заключение 29

Список использованных источников 31

Приложения

А – Основные процедуры desktop приложения 32

Б – Код класса MyDraw.java

ВВЕДЕНИЕ

Целью производственной практики по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем является закрепление и совершенствование приобретенных в процессе обучения профессиональных умений обучающихся по специальности, развитие общих и профессиональных компетенций, адаптация к конкретным условиям деятельности предприятий и организаций, освоение современных производственных процессов, обучение трудовым приемам, операциям и способам выполнения трудовых процессов.

В процессе прохождения производственной практики необходимо будет решить следующие задачи:

* приобрести практический опыт разработки алгоритма решения поставленной задачи и реализации его средствами автоматизированного проектирования, использования инструментальных средств на этапе отладки программного продукта и проведения тестирования программного модуля по определенному сценарию;
* приобрести практический опыт анализа алгоритмов, в том числе с применением инструментальных средств, разработки кода программного продукта на основе готовой спецификации на уровне модуля, осуществления рефакторинга и оптимизации программного кода;
* приобрести практический опыт разработки мобильных приложений;
* сформировать профессиональные умения и навыки;
* приобрести практические навыки в будущей профессиональной деятельности при выполнении функций дублера инженерно-технических работников;
* выполнить индивидуальное задание.

В результате прохождения производственной практики необходимо будет подтвердить уровень овладения видом деятельности «Разработка модулей программного обеспечения для компьютерных систем» и следующими профессиональными компетенциями:

ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием.

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4 Выполнять тестирование программных модулей.

ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода.

ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ.

Производственная практика проводится на базе ООО «Программист».

1 ХАРАКТЕРИСТИКА МЕСТА ПРАКТИКИ

ООО «Программист» - интернет магазин и удалённая поддержка. Компания была основана в 1997 году. Состоит из программистов, специалистов и консультантов, которые решают различные задачи связанные с 1С, онлайн-кассами, компьютерами и различными программами.

Юридический адрес ООО «Программист»: 385009, Россия, Республика Адыгея, г. Майкоп, ул. Юннатов, д. 28.

Основными задачами ООО «Программист» являются:

* поставка офисного и торгового оборудования;
* настройка, установка и продажа онлайн-касс;
* автоматизация бизнеса под ключ;
* консультация и обучение работе в системе ЕГАИС и других;
* выпуск электронных подписей;
* монтаж локальных сетей;
* создание и обслуживание web-сайтов;
* уполномоченная бухгалтерия;
* обслуживание и поддержка клиентов 1С;
* продажа лицензионного программного обеспечения;
* удалённая поддержка по всей России.

Оптовая торговля компьютерами, периферийными устройствами к компьютерам и программным обеспечениям. Организация работает более 23 лет на данном рынке.

Актуальная информация о компании представлена на сайте <https://prog01.ru>, а также в социальных сетях.

Предприятие работает с понедельника по пятницу с 09:00 до 17:00.

Руководитель организации: генеральный директор Мусиенко Виталий Владимирович.

Также, является руководителем отдела, в котором проходила практика – Мусиенко В.В.

2 СВЕДЕНИЯ ОБ ИМЕЮЩЕЙСЯ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКЕ И ПРОГРАММНОМ ОБЕСПЕЧЕНИИ

Практика проходила в техническом отделе предприятия.

Технический тодел имеет следующие характеристики:

* площадь (не менее 6 кв.м. на одно место) – 49,4 кв.м;
* электрощит – 1 шт.;
* электророзетки – 11 шт.;
* рабочие столы:
  + компьютерные – 7 шт.;
  + расстановка рабочих мест с ПЭВМ – периметральная.

В кабинете установлены 14 компьютеров со следующими характеристиками:

* материнская плата: Asrock H510M-HVS R2.0;
* процессор: 3.21 ГГЦ;
* видео карта: Intel HD Graphics 610;
* оперативная память (RAM): 8 ГБ;
* жесткий диск (SSD): 240 ГБ;
* тип привода: DVD+/-RW.

Компьютеры объединены в локальную сеть и имеют выход в Internet.

Программное обеспечение рабочего места:

* Microsoft Office 2019;
* Microsoft Visual Studio 2019 Community, включая следующие компоненты:
  + .NET desktop development Workload;
  + Universal Windows Platform development Workload;
  + Data storage and processing Workload;
  + Entity Framework (EF); IIS, ASP.NET and web development, Mobile development with .NET;
* Android Studio 4.2.1, включая следующие компоненты:
  + Android SDK Tools;
  + Android SDK Platform-Tools;
  + Android SDK Build-Tools;
  + Android SDK Platform;
  + USB Driver;
* Android SDK 30.29;
* SQL Server Management Studio 18.9.1;
* Visual Studio Code 1.57;
* Notepad++ 8.1;

Программное обеспечение сервера:

* Microsoft Visio Professional 2019;
* Microsoft Office 2019;
* .NET Framework Developer Pack 4.8;
* Microsoft Visual Studio 2019 Community, включая следующие компоненты:
  + .NET desktop development Workload;
  + Universal Windows Platform development Workload;
  + Data storage and processing Workload;
  + Entity Framework (EF); IIS, ASP.NET and web development, Mobile development with .NET;
* Android Studio 4.2.1, включая следующие компоненты:
  + Android SDK Tools;
  + Android SDK Platform-Tools;
  + Android SDK Build-Tools;
  + Android SDK Platform;
  + USB Driver;
* Android SDK 30.29;
* SQL Server Management Studio 18.9.1;
* Visual Studio Code 1.57;
* Notepad++ 8.1;
* Microsoft SQL Server 2017 Express;

3 СВЕДЕНИЯ О ПРОХОЖДЕНИИ ПРАКТИКИ В КАЧЕСТВЕ ДУБЛЕРА ИНЖЕНЕРНО-ТЕХНИЧЕСКОГО РАБОТНИКА

Во время прохождения практики в качестве дублера техника-программиста я выполнял следующие действия:

* ознакомление с системным и прикладным программным обеспечением на предприятии;
* работа с документацией;
* сбор и структурирование материала по теме индивидуального задания «Информационно-справочная система “Психологические тесты”;
* анализ алгоритмов, в том числе с применением инструментальных средств;
* разработка кода программного продукта на основе готовой спецификации на уровне модуля;
* написание программного кода, реализующего функциональность разрабатываемого приложения;
* разработка интерфейса приложения;
* отладка программного продукта с использованием спе­циализированных программных средств;
* разработка тестовых наборов и тестовых сценариев;
* осуществление рефакторинга и оптимизации программного кода;
* тестирование программного продукта;
* разработка технической документации к программному продукту;
* обслуживание оргтехники;
* установка и настройка необходимого программного обеспечения;
* разработка кода по заданию руководителя практики.

Выполненная работа соответствует профессиональным компетенциям, предусмотренным программой производственной практики ПМ.01 специальности 09.02.07 Информационные системы и программирование.

4 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

4.1 Постановка задачи

В период практики необходимо написать desktop-приложение с психологическими тестами, и возможностью узнать способы обращения с проблемами к психологу/психотерапевту, а также версию данного приложения для мобильных платформ. Программа должна содержать текстовую и графическую информацию, достаточную для исследования своего психологического здоровья, иметь удобный и понятный пользователю графический интерфейс, быть проста в использовании.

Программа должна предлагать следующие возможности:

* возможность пройти психологический тест с вариантами ответов;
* узнать результат;
* строить график функции;
* ;
* выдавать информацию о приложении.

Входными данными будут:

* результаты исследования функции y = cosec(А\*x);
* интервал, на котором необходимо построить график функции;
* значение параметра;
* выбранный масштаб.

Выходными данными будут:

* визуальное отображение графика функции;
* информация, полученная при исследовании функции;
* пользовательская справочная система.

4.2 Проектирование структуры программы

4.2.1 Desktop версия приложения

Разработанное приложение имеет следующие окна:

* загрузочное окно;
* главное окно, которое содержит:
* о;
* окно «О программе» и сведения о разработчике;
* окно;
* справка;

4.2.2 Мобильная версия приложения

Разработанное приложение имеет следующие окна:

* загрузочное окно;
* главное окно, которое содержит:
* окно «О программе» и сведения о разработчике;
* логотип приложения;
* кнопки для перехода на тесты;
* окно с информацией о вариантах психологической помощи;
* окна с информацией о каждом тесте;
* окна с вопросами теста;
* окно результатов.

4.3 Функциональное взаимодействие модулей программы

4.3.1 Desktop версия приложения

На основе разработанного технического проекта была написана информационно-справочная система «Психологические тесты». Для создания приложения была использована среда программирования Visual Studio 2019 и язык программирования С#. В программе используются следующие языковые модули:

* System.Windows.Controls− предоставляет классы для создания элементов, известных как элементы управления, позволяющих пользователю взаимодействовать с приложением;
* System.Diagnostics− содержит типы, позволяющие взаимодействовать с системными процессами, журналами событий и счётчиками производительности;
* System− содержит фундаментальные и базовые классы, определяющие часто используемые типы значений и ссылочных данных, события и обработчики событий, интерфейсы, атрибуты и исключения обработки;
* System.Windows− содержит типы, используемые в приложениях WindowsPresentationFoundation (WPF), включая клиенты анимации, элементы управления пользовательского интерфейса, привязку данных и преобразование типов;
* CefSharp− содержит типы, поддерживающие компиляцию и создание кода на языке C#, а также поддерживающие взаимодействие между средой DLR и языком C#;
* System.IO −cодержит типы, поддерживающие ввод и вывод, включая возможности чтения и записи данных в потоках как синхронно, так и асинхронно, реализации пользовательского ведения журнала и обработки входящих и исходящих потоков данных в последовательных портах;
* System.Data− содержит типы для доступа к данным из различных источников и для управления этими данными. Пространство имен верхнего уровня и несколько дочерних пространств имен образуют архитектуру ADO.NET и поставщиков данных ADO.NET [9].
* System.Threading− содержат типы, обеспечивающие возможности многопотокового программирования. Дочернее пространство имен предоставляет типы, которые упрощают задачу написания параллельного и асинхронного кода;
* System.Collections.Generic – работа с обобщёнными коллекциями (List<string>) [5].

В программе были созданы следующие пользовательские модули:

* Zastavka.cs (окно заставки) – заставка приложения;
* MainForm.cs (главное окно программы) − обеспечивает функциональность для построения графика функции, ввода диапазона построения и параметров, масштабирования, а также обеспечивает доступ к системе помощи;
* Help.cs – система помощи, которая содержит справочную информацию о программе, а также инструкцию по работе с ней;
* Oprog.cs (окно о программе) – этот модуль содержит сведения о разработчике программы.

В модуле MainForm.cs описаны и используются следующие процедуры:

* private void ShowButton\_Click(object sender, EventArgs e) − производит построение графика функции в поле для построения;
* private void HelpButton\_Click(object sender, EventArgs e) − открывает справку;
* private void ScaleButton\_Click(object sender, EventArgs e) – обеспечивают масштабирование графика.
* …

Код данных процедур представлен в приложении А.

4.3.2 Мобильная версия приложения

На основе разработанного технического проекта была написана мобильная версия информационно-справочной системы «Психологические тесты». Для создания приложения была использована среда программирования Android Studio и язык программирования Kotlin. В программе используются следующие языковые модули:

* android.content.Context - это базовый абстрактный класс, реализация которого обеспечивается системой Android. Этот класс имеет методы для доступа к специфичным для конкретного приложения ресурсам и классам и служит для выполнения операций на уровне приложения, таких, как запуск активностей, отправка широковещательных сообщений, получение намерений и прочее;
* android.graphics.Color - содержит несколько констант и методов для конвертации и извлечения цветового компонента;
* android.view.View – компонент (виджет), который рисуется на экране;
* android.content.Intent - класс, объекты которого используются для взаимодейтвия между отдельными частями Android приложения;
* android.os.Bundle - необходим для временного хранения данных в процессе выполнения;
* android.widget.Toast - это небольшое, всплывающее сообщение, появляющееся внизу и экрана на непродолжительное время;
* androidx.appcompat.app.AppCompatActivity – это базовый класс для действий, которые используют функциональные возможности панели поддержки библиотеки;
* android.widget.TextView – это стандартный виджет, предназначенный для отображения текста без возможности редактирования его пользователем;
* android.os.Handler – это класс, позволяющий работать с очередью сообщений.

В программе были созданы следующие пользовательские модули:

* SplashScreenActivity.kt – загрузочное окно;
* HomeActivity.kt – главное окно, в котором находится все основные элементы и компоненты приложения;
* InfoActivity.kt – окно с информацией о приложении и сведений о разработчике;
* HelpUserActivity.kt – окно с информацией о возможной психологической помощи;
* SelfEsteemScreen.kt – окно с информацией о тесте «Тест на самооценку»;
* GoodPersonScreen.kt - окно с информацией о тесте «Вы добрый человек?»;
* HealthScreen.kt – окно с информацией о тесте «Хорошее ли у Вас психологическое здоровье?»;
* SelfEsteemTest.kt – окно для теста «Тест на самооценку»;
* GoodPersonTest.kt – окно для теста «Вы добрый человек?»;
* HealthTest.kt – окно для теста «Хорошее ли у Вас психологическое здоровье?»;
* ResultSelfEsteemTest.kt – окно результатов теста «Тест на самооценку»;
* ResultGoodPersonTest.kt – окно результатов тесты «Вы добрый человек?»;
* ResultHealthTest.kt – окно результатов тесты «Хорошее ли у Вас психологическое здоровье?»;
* setDataSelfEsteem.kt – объект, в котором хранятся вопросы для теста «Тест на самооценку»;
* setDataGoodPerson.kt – объект, в котором хранятся вопросы для теста «Вы добрый человек?»;
* setDataHealth.kt – объект, в котором хранятся вопросы для теста «Хорошее ли у Вас психологическое здоровье?»;
* QuestionsData.kt – класс данных, который хранит все необходимые идентификаторы теста.

В модуле SplashScreenActivity.kt описаны и используются следующие процедуры:

* supportActionBar?.hide() – метод для скрытия панели в верхней части приложения;
* Handler().postDelayed() – таймер для загрузочного окна;

В модуле HomeActivity.kt описаны и используются следующие процедуры:

* supportActionBar?.hide() – метод для скрытия панели в верхней части приложения;
* override fun onBackPressed() – функция, позволяющая выйти из приложения двойным нажатием кнопки «Назад»;
* fun goSelfEsteemTest() – переход на другую активность;
* fun goGoodPersonTest() – переход на другую активность;
* fun btnInfo() – переход на другую активность;
* fun onHelperUserClick() – переход на другую активность;
* fun goHealthTest() – переход на другую активность;

В модуле Activity\_ChartSettings.java описаны и используются следующие процедуры:

* public void onClickExit(View v) – обработка нажатия кнопки выхода без сохранения;
* public void onClickAcceptExit(View v) - обработка нажатия кнопки выхода с сохранением;
* public void saveText() – метод сохранения данных.

В модуле MyDraw.java описаны и используются следующие процедуры:

* public MyDraw(Context context) – конструктор класса MyDraw;
* protected void onDraw(Canvas canvas) – метод, который вызывает методы для отрисовки холста и графика;
* private void drawMetric(Canvas canvas) – метод для отрисовки системы координат и холста;
* private void drawChart(Canvas canvas) – метод для отрисовки графика.

Код класса MyDraw.java представлен в приложении Б.

4.5Работа с приложением

4.5.1 Desktop версия приложения

При запуске программы появляется заставка (рисунок 1).

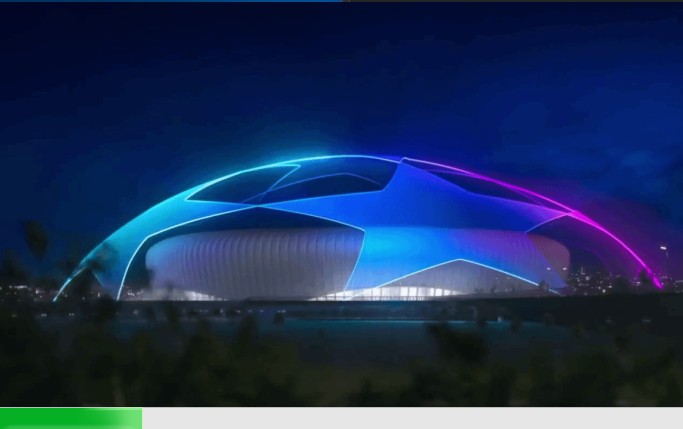


Рисунок 1 – Заставка

После завершения экрана загрузки, открывается Главное окно программы.

Интерфейс программы имеет следующие элементы (рисунок 2):

* область вывода графика функции (1);
* кнопка вызова справки (2);
* кнопка, закрывающая приложение (3);
* поле для ввода начального значения диапазона построения (4);
* поле для ввода конечного значения диапазона построения (5);
* кнопка, по нажатию на которую строится график (6);
* поле текущего масштаба функции (7);
* кнопки, позволяющие изменить значение масштаба функции (8);
* информация о графике (9).

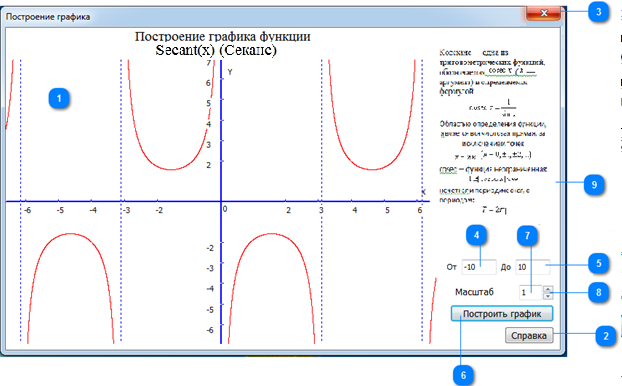


Рисунок 2 – Основное окно программы

Для исследования функции нужно указать левую и правую границы построения графика и масштаб. Далее надо нажать кнопку «Построить график». После этого на экране появятся уравнение функции и график. При необходимости можно изменить параметры.

Если у пользователя возникнуть вопросы по работе с программой, он может вызвать справочное окно, нажав на кнопку «Справка» или выбрав пункт меню «Справка» в Главном окне (рисунок 3).

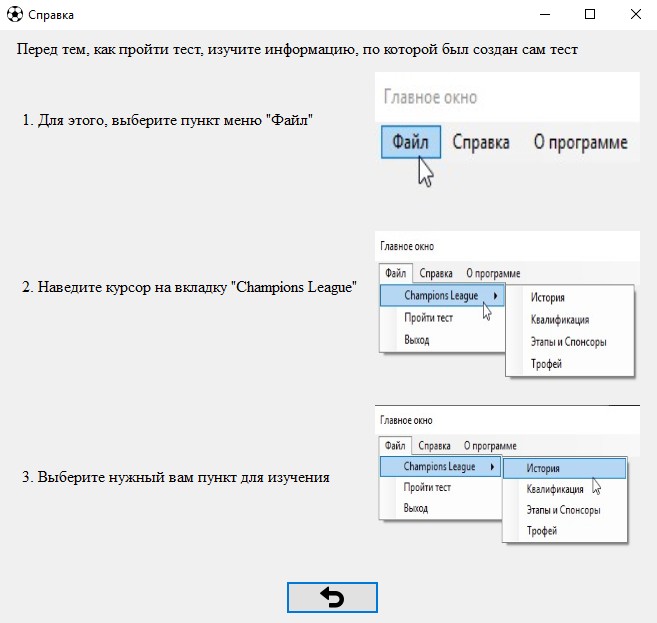


Рисунок 3 – Окно справки

При нажатии пункта меню «О программе» в главном окне, откроется окно просмотра информации о создателе программы (рисунок 4).

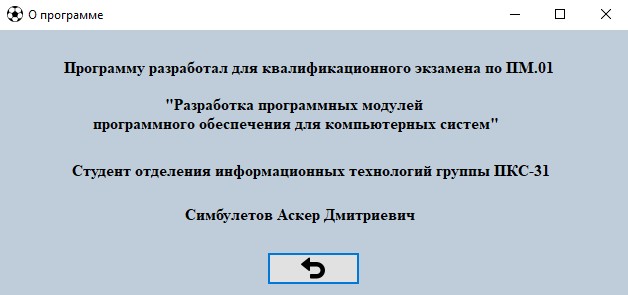


Рисунок 4 – Окно о программе

4.5.2 Мобильная версия приложения

При запуске программы открывается заставка (рисунок 5).

После завершения загрузки, откроется главное окно программы.

Меню приложения появится после нажатия на кнопку 1 (рисунок 6):

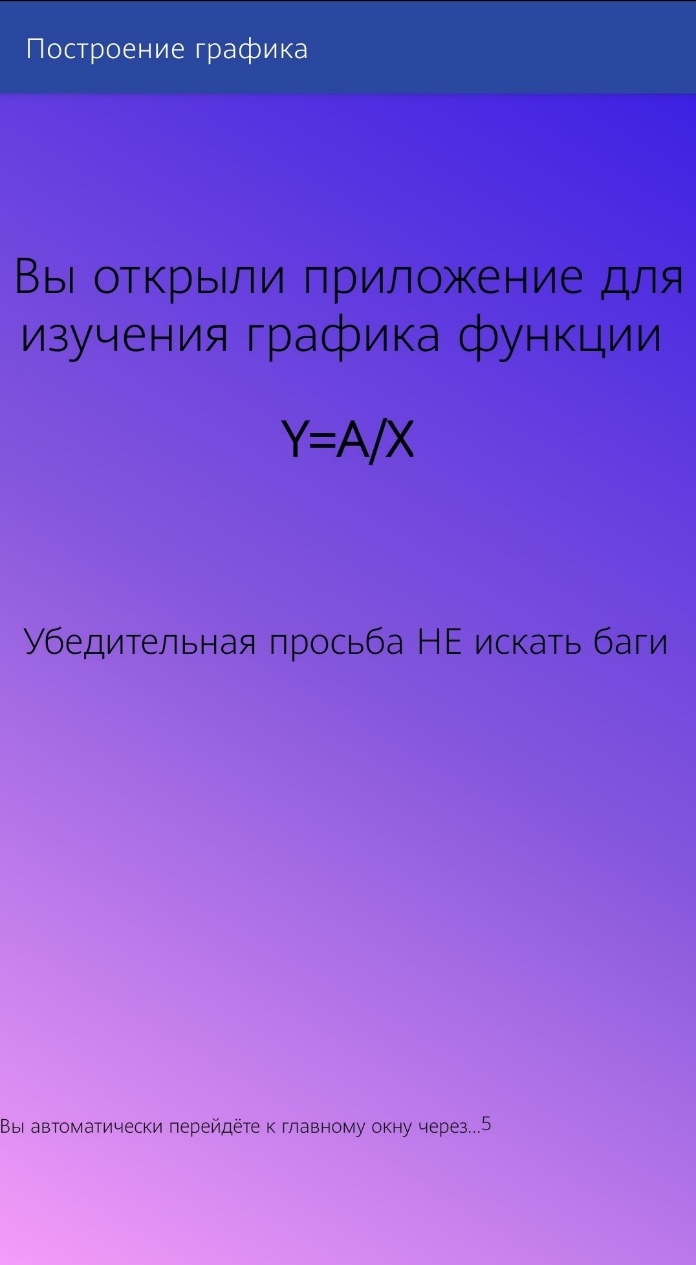


Рисунок 5 – Заставка

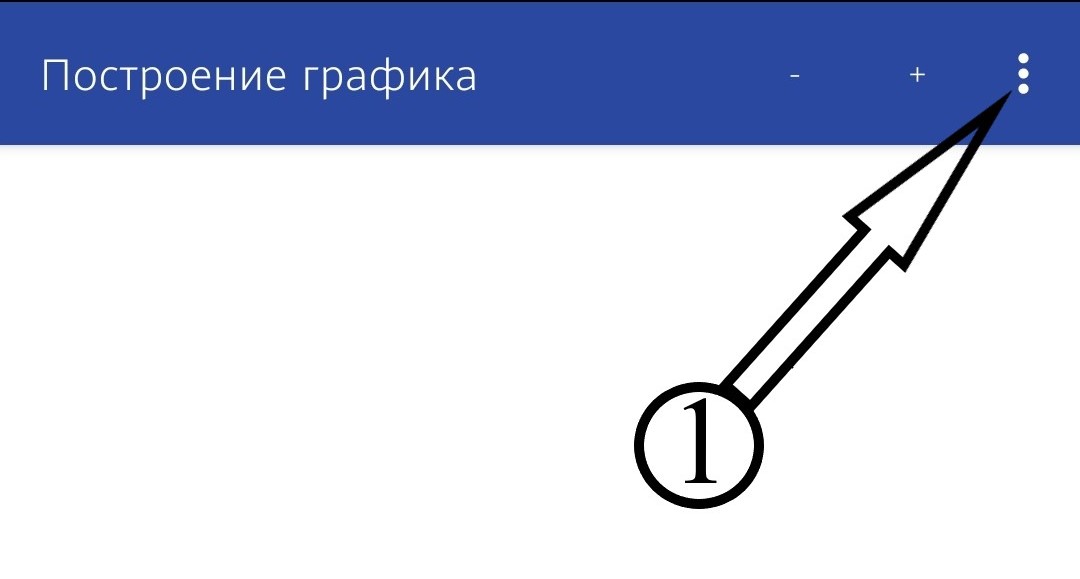


Рисунок 6 – Основное окно программы

Далее появится список с функциями приложения (рисунок 7).

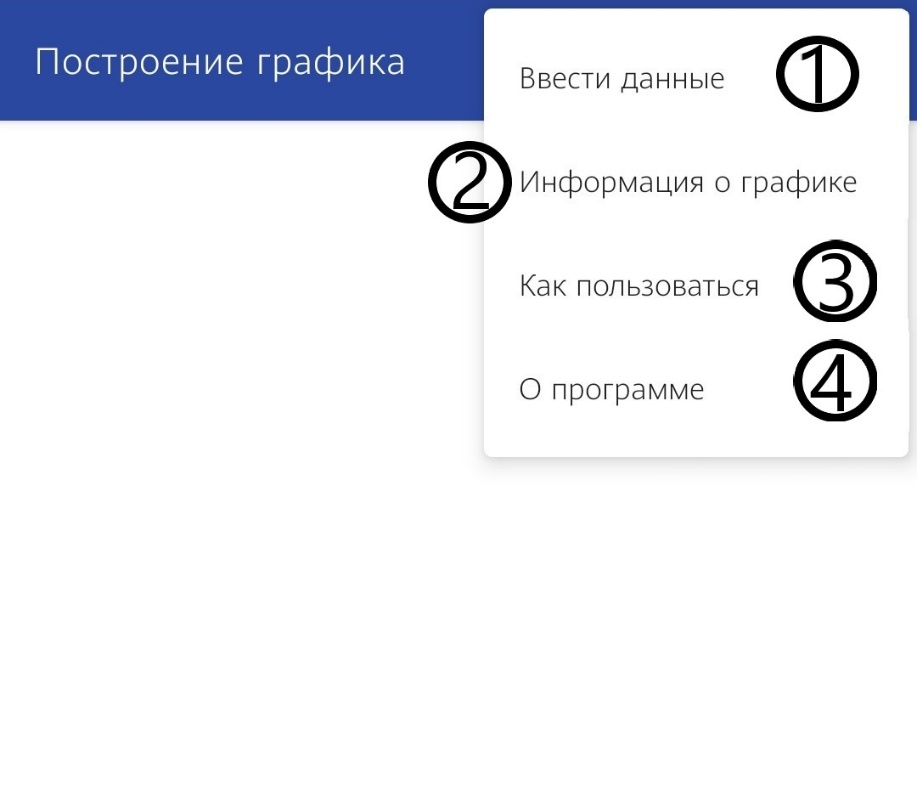


Рисунок 7 – Меню приложения

Для исследования функции необходимо нажать «Ввести данные» (кнопка 1, рисунок 7), указать левую и правую границы построения графика и параметр функции. Далее надо нажать кнопку «Принять». Если вы решили не принимать значения, то нажмите кнопку «Назад». После этого на экране появится график. Для отображения информации о графике нажмите кнопку «Информация о графике» (кнопка 2, рисунок 7), затем появится всплывающее окно (рисунок 8).

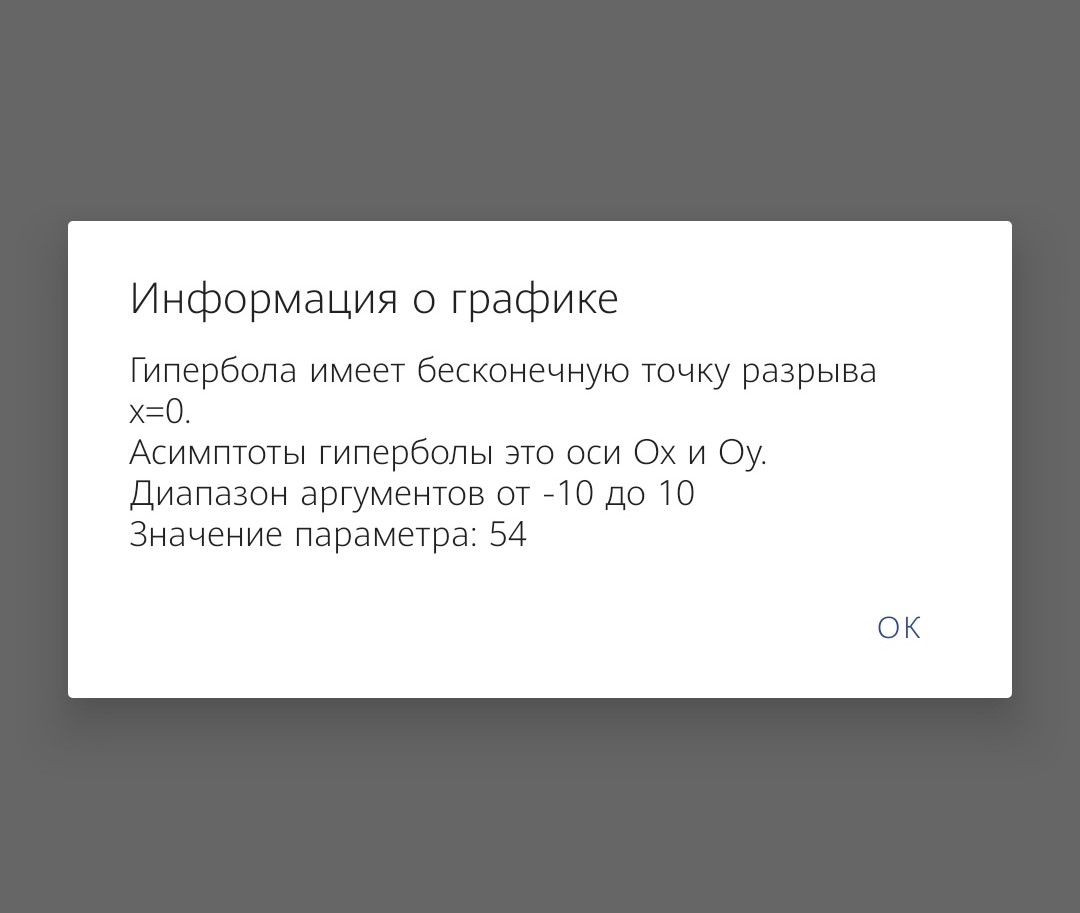


Рисунок 8 – Окно информации о графике

Если у пользователя возникнуть вопросы по работе с программой, он может вызвать справочное окно, выбрав пункт меню «Как пользоваться» (кнопка 3, рисунок 7) в главном окне (рисунок 9).

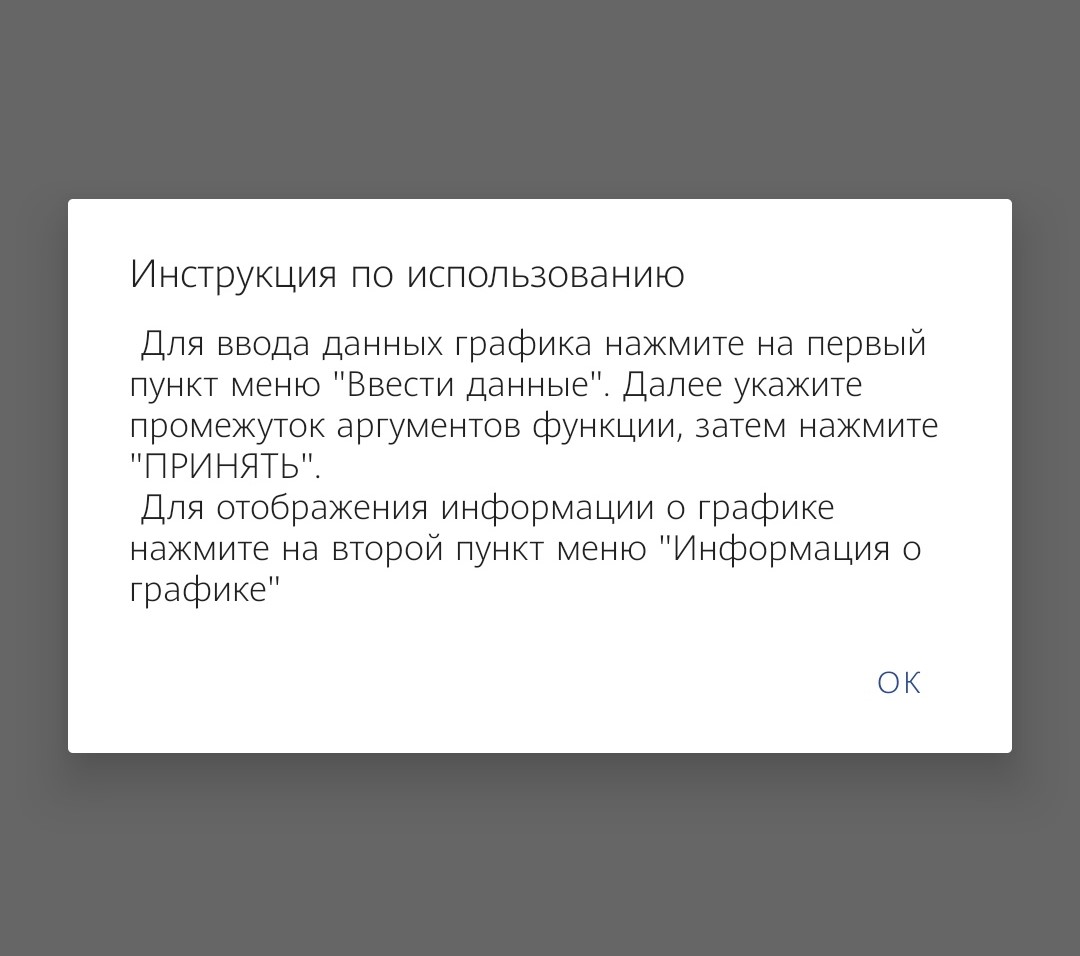


Рисунок 9 – Окно инструкции по использованию

При нажатии пункта меню «О программе» (кнопка 4, рисунок 7) в главном окне, откроется окно просмотра информации о создателе программы (рисунок 10).

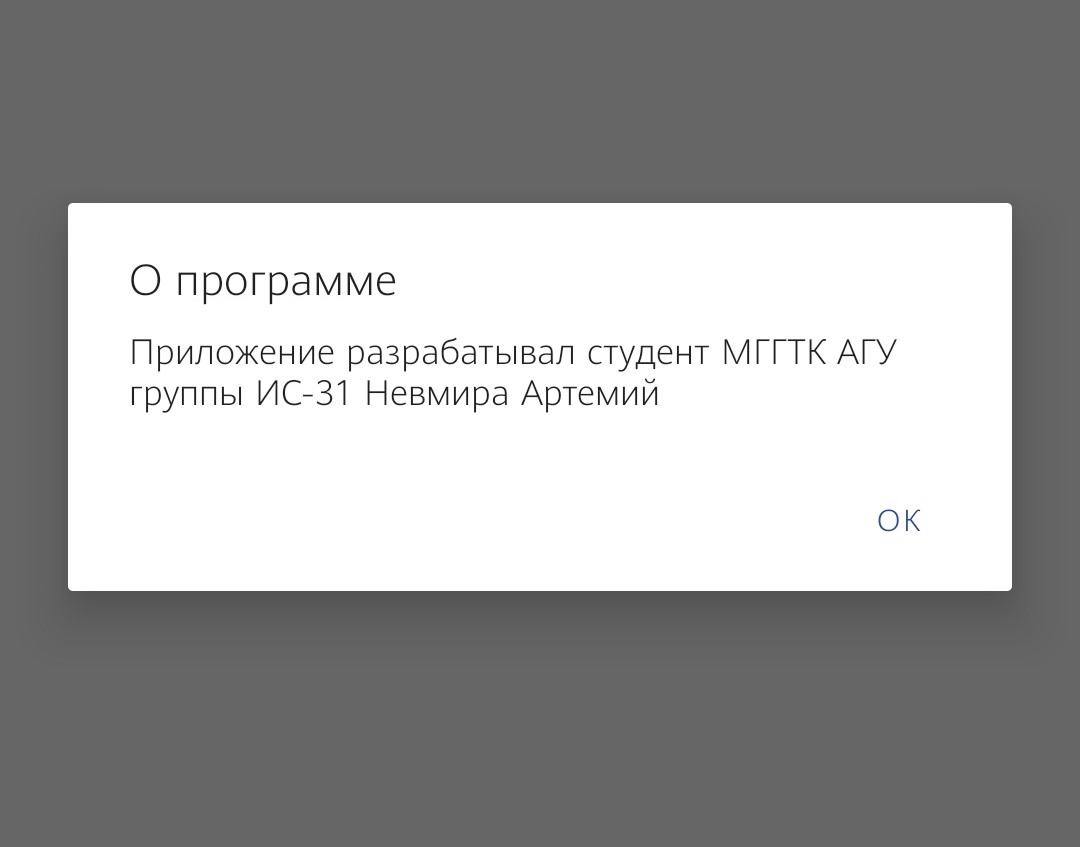


Рисунок 10 – Окно о программе

ЗАКЛЮЧЕНИЕ

В результате прохождения производственной практики мною были закреплены теоретические знания, полученные при изучении ПМ.01; отработаны профессиональные компетенции по получаемой специальности; приобретены практические навыки в будущей профессиональной деятельности; выполнено индивидуальное задание.

Мною было разработано приложение для исследования функции   
y = cosec(A\*x).

Программа имеет удобный и понятный пользователю графический интерфейс, проста в использовании и обеспечивает следующие возможности:

* ввод и вывод диапазона построения функции;
* ввод значения параметра функции;
* строит график функции;
* масштабирует график функции;
* выдает справочную информацию;
* сохранять результаты исследования в файлах.

В ходе выполнения разработки данного приложения были:

* проанализирована методика исследования функции;
* спроектирована структура приложения и программного кода;
* разработаны справочные материалы;
* выбраны средства реализации приложения;
* разработан программный код приложения;
* разработан тестовый проект, реализующий оптимальный набор unit-тестов.

Основными достоинствами программы являются:

* программа проста в установке, настройке и эксплуатации;
* данный программный продукт предназначен для широкого круга пользователей;
* наличие справочной системы.

К недостаткам данной программы можно отнести то, что она предназначена для исследования только одной функции.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Албахари Б., Албахари Дж. С# 7.0. Справочник. Полное описание языка. М.: Вильямс, 2018. 1024 с.
2. Виссер Д. Разработка обслуживаемых программ на языке С#. М.: ДМК Пресс, 2017. 194с.
3. Дарвин Я. Ф. Android. Сборник рецептов: задачи и решения для разработчиков приложений. 2-е изд. Пер. с англ. СПб.: ООО «Альфа-книга», 2018. 768 с.
4. Джепикс Ф., Троелсен Э. Язык программирования C# 7 и платформы .NET и .NET Core. М.: Вильямс, 2018. 1328 с.
5. Мюллер Д. П., Семпф Б., Сфер Ч. С# Для чайников. М.: Диалектика, 2016. 608 с.
6. Скит Д. С# Для профессионалов. Тонкости программирования. М.: Вильямс, 2019. 608 с.
7. Документация Microsoft. URL: https://docs.microsoft.com/ru-ru/documentation (дата обращения: 01.12.2021).
8. Библиотека классов .NET Framework. URL: https:// msdn.microsoft.com/ru-ru/library/mt472912(v=vs.110).aspx (дата обращения: 01.12.2021).
9. Как пользоваться timer в c#. URL: https://docs.microsoft.com/ru-ru/dotnet/api/system.timers.timer?view=netcore-2.1 (дата обращения: 05.12.2021).
10. Космос Планет. Сайт всё о космосе. URL: https://cosmosplanet.ru/ solnechnayasistema/stroenie-solnechnoj-sistemy.html (дата обращения: 02.12.2021).
11. Общие сведения о платформе Android. URL: https://developer.android.com/guide (дата обращения: 30.11.2021).

ПРИЛОЖЕНИЕ А

Основные процедуры desktop приложения