

УДК 004.021:004.43

ПРИМЕНЕНИЕ ЯЗЫКА ПРОГРАММИРОВАНИЯ C++ ДЛЯ РАЗРАБОТКИ ЭЛЕКТРОННЫХ ОБРАЗОВАТЕЛЬНЫХ РЕСУРСОВ ПО МАТЕМАТИКЕ

Ведерников Н.А., Бужинская Н.В.

Нижнетагильский государственный социально-педагогический институт, филиал ФГАОУ ВО «Российский государственный профессионально-педагогический университет», Нижний Тагил, e-mail: oloistcsgo@yandex.ru

Программирование является достаточно сложным для усвоения. Кроме того, часть учащихся не заинтересована в его изучении, несмотря на то, что задания по программированию включены в Единый государственный экзамен. Следовательно, учителю информатики необходимо найти методы для развития интереса учащихся к решению задач по программированию. Одним из способов реализации данной цели является демонстрация возможностей программных продуктов, разработанных с помощью языков программирования. Возможности языков программирования позволяют создавать электронные образовательные ресурсы соответственно методическим целям. В статье рассматривается вопрос разработки виртуальной лаборатории по математике с помощью высокоуровневого языка программирования C++. Область применения C++ позволяет создавать на нем драйверы, операционные системы, прикладные программы. Представленный в статье ресурс предназначен для построения графиков математических функций в зависимости от задаваемых параметров. Кроме того, ресурс позволяет увеличивать область построения графика для его более детального изучения и сохранить эту область в отдельном файле. Интерфейс виртуальной лаборатории был разработан в программе QT Designer. Его можно использовать для проведения уроков, как по математике, так и по информатике.

Ключевые слова: электронный образовательный ресурс, C++, математическая лаборатория, график функции, код, интерфейс

APPLICATION OF C ++ PROGRAMMING LANGUAGE FOR DEVELOPMENT OF ELECTRONIC EDUCATIONAL RESOURCES IN MATHEMATICS

Vedernikov N.A., Buzhinskaya N.V.

Nizhny Tagil State Socio-Pedagogical Institute (branch) FSAEI of HE «Russian State Vocational Pedagogical University», Nizhny Tagil, e-mail: oloistcsgo@yandex.ru

Programming is difficult enough to learn. In addition, some students are not interested in studying it, despite the fact that programming tasks are included in the Unified state exam. Therefore, a computer science teacher needs to find methods to develop students' interest in solving programming problems. One way to achieve this goal is to demonstrate the capabilities of software products developed using programming languages. Possibilities of programming languages allow to create electronic educational resources according to methodical purposes. The article deals with the development of a virtual laboratory in mathematics using a high-level C++ programming language. The scope of C++ allows you to create drivers, operating systems, applications on it. The resource presented in the article is intended for plotting mathematical functions depending on the specified parameters. In addition, the resource allows you to increase the area of plotting for its more detailed study and save this area in a separate file. The interface of the virtual laboratory was developed in QT Designer. It can be used for lessons, both in mathematics and computer science.

Keywords: electronic educational resource, C++, mathematical laboratory, function graph, code, interface

В настоящее время развитие информационно-коммуникационных технологий приводит к изменению роли учителя. Учащиеся получают большой объем информации из интернета, а учитель должен научить их методам и способам обработки, структурирования и анализа данных. Организация урока с помощью современных программных и/или технических средств всегда вызывает повышенный интерес обучающихся. В данной статье мы рассмотрим особенности разработки электронных образовательных ресурсов.

Под электронными образовательными ресурсами (ЭОР) понимаются документы, которые представлены в электронной форме и служат для реализации целей учебного процесса [1]. Электронные образовательные ресурсы являются важнейшим компонентом цифрового образовательного пространства.

Выделяются различные классификации электронных образовательных ресурсов. Они могут быть разделены по целевому назначению (для учителя, учащихся, родителей), по типу решаемых задач (для контроля знаний, объяснения нового материала, организации досуга), по тематике, индивидуальным характеристикам обучающихся, по характеру предоставления информации и т.д. При этом интернет предоставляет свободный доступ к большому количеству электронных образовательных ресурсов, разработанных другими педагогами. В этом случае задачей учителя является грамотный отбор таких электронных образовательных ресурсов и их адаптация под собственную методику. Кроме того, при использовании готовых ЭОР следует помнить об авторских правах разработчика.

Отметим, что, на наш взгляд, наибольшую методическую значимость приобретают электронные образовательные ресурсы, которые могут одновременно выступать и средством обучения, и предметом исследования. В этом случае разработка электронного образовательного ресурса является достаточно трудоемким процессом и требует дополнительных временных затрат для учителя. Например, к таким электронным образовательным ресурсам можно отнести программные продукты, разработанные с помощью языков программирования. В этом случае учитель может использовать разработанный программный продукт как для реализации функций урока, так и для демонстрации возможностей того или иного языка программирования.

Кроме того, решение многих задач по программированию предполагает наличие минимально необходимой суммы математических знаний и умений. На протяжении долгого времени между специалистами IT-сферы ведется спор, какую роль математика играет в программировании. В настоящее время ответ может быть однозначным. Развитие искусственного интеллекта, изучение нейронных сетей, разработка криптографических методов свидетельствует о том, что знания математики необходимы для анализа информации, прогнозирования и решения задач. Поэтому возникает необходимость показать учащимся важность математики в программировании [2].

Материалы и методы исследования

Для решения данной задачи нами был разработан электронный образовательный ресурс по математике «Математическая лаборатория».

Процесс разработки данного ресурса занял три месяца. В таблице представлены основные этапы разработки и продолжи-

тельность каждого из них. Одним из самых важных и длительных этапов являются изучение требований к данному ресурсу, изучение средств разработки и непосредственно сам этап разработки.

К основным рискам, которые могли повлиять на сроки реализации и качество проекта, можно отнести:

- дополнительные затраты времени на повышение уровня знаний в области математики;

- трудности с разработкой лаборатории на языке C++;

- необходимость учета индивидуальных особенностей учащихся 9–11 классов при разработке ресурса.

Для определения функционала данного ресурса был проведен опрос учащихся 9-го класса МОУ СОШ № 80 г. Нижнего Тагила. Им был задан вопрос: «Какая из тем математики вызывает у вас наибольшие затруднения?». Ответы респондентов представлены на рис. 1.

Как показывают результаты опроса, наибольшие затруднения у учащихся вызывают темы «Числовые функции» и «Элементы теории тригонометрических функций». Таким образом, было принято решение разработать электронный образовательный ресурс, который позволяет работать с графиками тригонометрических функций – строить их в зависимости от задаваемых параметров, изменять интервал, устанавливать цвет и толщину линий. Разработка подобного ресурса позволит решить две важнейшие задачи:

- продемонстрировать возможности программирования для реализации проектов;

- показать работу с графиками функции.

Для более детального определения функционала данного продукта была построена диаграмма прецедентов, представленная на рис. 2.

Таблица 1

Этапы разработки проекта

№ п/п	Этап реализации	Начальная дата	Конечная дата	Продолжительность этапа
1	Определение требований к программному продукту	20.10.2018	28.10.2018	8
2	Анализ результатов опросов	22.10.2018	23.10.2018	1
3	Определение функционала программного продукта	24.10.2018	30.10.2018	6
4	Выбор средств для разработки программного продукта	31.10.2018	05.11.2018	5
5	Изучение средств разработки программного продукта	25.10.2018	10.11.2018	16
6	Проектирование пользовательского интерфейса	25.10.2018	27.10.2018	2
7	Разработка программного продукта	27.10.2018	25.11.2018	29
8	Тестирование программного продукта	27.10.2018	05.11.2018	9
9	Анализ результатов тестирования	05.11.2018	10.11.2018	5
10	Внедрение программного продукта в деятельность ОУ	25.10.2018	10.11.2018	16
11	Анализ результатов применения программного продукта	28.11.2018	29.11.2018	1

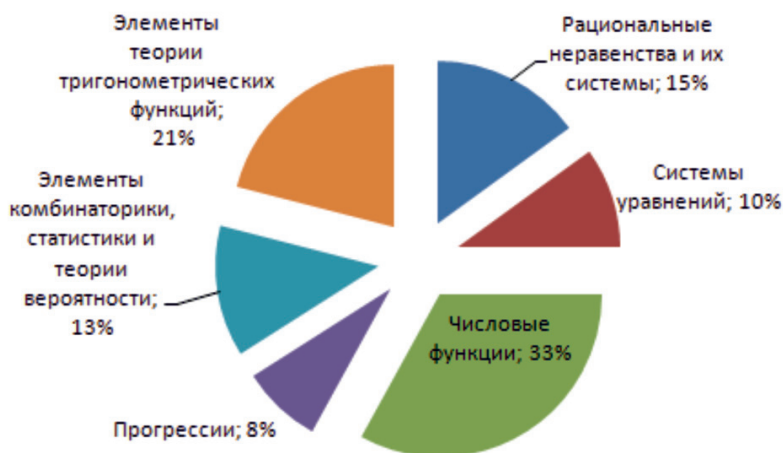


Рис. 1. Результаты опроса учащихся



Рис. 2. Диаграмма прецедентов, отражающая требования к лаборатории

Учащийся (пользователь) может выбрать функцию для более детального изучения, а затем построить график функции в заданной плоскости. Важной особенностью данного ресурса является возможность сохранения построенного графика и/или выделения его отдельного фрагмента.

Для разработки лаборатории использовался язык программирования C++. Выбор данного языка объясняется возможностью использовать объектно-ориентированный подход, писать обобщенный код с помощью шаблонов, достаточно простой обработки ошибок за счет исключений.

Визуальная составляющая была разработана в программе *QtDesigner* – инструменте для проектирования и создания графических пользовательских интерфейсов (GUI) из компонентов Qt [3–5].

Рассмотрим принцип работы лаборатории. Чтобы начертить график, необходимо задать параметры начального и конечного значений плоскости, на которой будет начертан сам график. После нужно выбрать функцию, которую нужно начертить, и нажать на кнопку «Нарисовать». На рис. 3 представлен интерфейс приложения, в листингах 1, 2 – программная реализация.

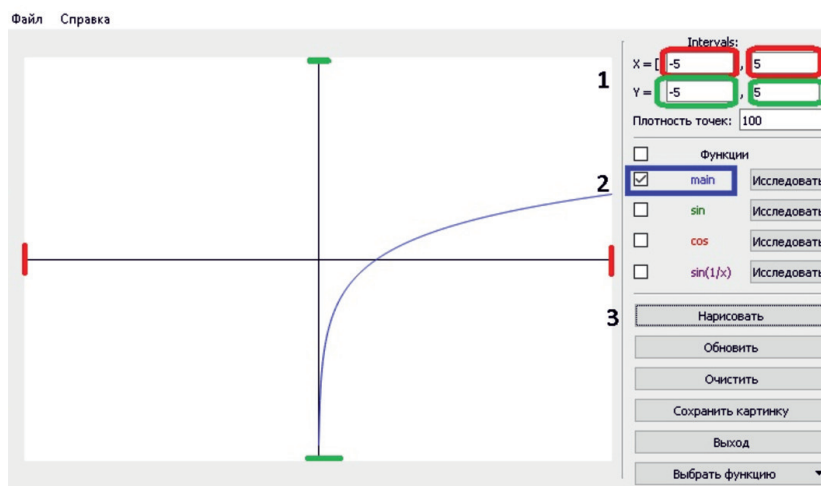


Рис. 3. Интерфейс виртуальной лаборатории

При нажатии кнопки «Нарисовать» в функции `MainWindow::on_draw_clicked()` срабатывает несколько инкапсулированных функций, необходимых для работы лаборатории.

1. Функция `void MainWindow::getData();` // собирает всю информацию, которая была введена пользователем.

Листинг 1

```
leftX = ui->inputLeftX->text().toDouble();
leftX_af = ui->inputLeftX->text().toDouble();
rightX = ui->inputRightX->text().toDouble();
rightX_af = ui->inputRightX->text().toDouble();
leftY = ui->inputLeftY->text().toDouble();
leftY_af = ui->inputLeftY->text().toDouble();
rightY = ui->inputRightY->text().toDouble();
rightY_af = ui->inputRightY->text().toDouble();
step = 1.0/ui->inputAccuracy->text().toDouble();
```

2. Функция `void MainWindow::recountPixels();` // определяет размер одного пикселя внутри программы.

Листинг 2

```
onePixelX = 540.0/(rightX - leftX);
onePixelY = 370.0/(rightY - leftY);
Ox = fabs(leftX); Oy = rightY;
onePixelX_af = 540/(rightX_af - leftX_af);
onePixelY_af = 370/(rightY_af - leftY_af);
Ox_af = fabs(leftX_af); Oy_af = rightY_af;
```

3. Функция `void MainWindow::drawGraph();` // непосредственно рисует сам график по тем данным, которые ранее определили предыдущие вызванные функции.

Кроме того, программа позволяет строить сразу несколько графиков функций (рис. 4).

При нажатии кнопки «Нарисовать» в функции `MainWindow::on_draw_clicked()` срабатывает несколько инкапсулированных функций:

– `void MainWindow::getData();` // собирает всю информацию, которая была введена пользователем;

– `void MainWindow::recountPixels();` // определяет размер одного пикселя внутри программы;

– `void MainWindow::drawGraph();` // непосредственно рисует сам график по тем данным, которые ранее определили предыдущие вызванные функции.

При нажатии кнопки «Очистить» срабатывает метод `void MainWindow::on_clear_clicked()`. Данный метод вызывает в себе внутренние функции:

– `void MainWindow::clearData();` // обнуляет все переменные

– `void MainWindow::drawGraph(false);` // с данным параметром метод не будет рисовать графики, а оставит только координатную плоскость.

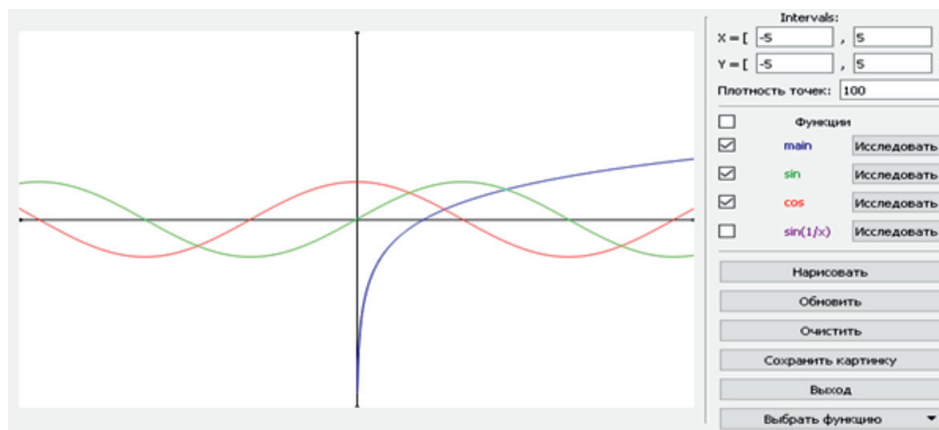


Рис. 4. Построение нескольких графиков функций

Кнопка «Сохранить картинку» сохраняет изображение графика в заранее установленном месте. Классы *QTime* и *QDate* хранят в себе время и дату в реальном времени соответственно. *QString::number()* конвертирует числовые значения в строковые. Когда объект времени методом *void QTime::hour()* возвращает значение часа в числовом типе данных, то данный строковый метод превращает его в строку. Эту строку можно конкатенировать с общей строкой, которая затем присваивается объекту типа *QMessageBox*. Если же значение времени не двухзначное, то перед ним ещё добавляется «0». Реализация представлена в листинге 4.

Листинг 4

```
void MainWindow::on_save_clicked()
{
    QTime time = QTime::currentTime();
    QDate date = QDate::currentDate();
    QString name;
    if(date.day() < 10)
        name += "0";
    name += QString::number(date.day()) + ".";
    if(date.month() < 10)
        name += "0";
    name += QString::number(date.month()) + ".";
    name += QString::number(date.year()) + "_";
    if(time.hour() < 10)
        name += "0";
    name += QString::number(time.hour()) + "-";
    if(time.minute() < 10)
        name += "0";
    name += QString::number(time.minute()) + "-";
    if(time.second() < 10)
        name += "0";
    name += QString::number(time.second());
    QFile file(name + ".png");
    qDebug() << name;
    file.open(QIODevice::WriteOnly);
    QMessageBox msgBox;
    msgBox.setStandardButtons(QMessageBox::Ok);
    if(ui->outputGraph->pixmap()->save(&file, "PNG")) {
        msgBox.setText(«Saved to program folder with name: "+name+".png");
        msgBox.setWindowTitle("Saved!");
    }
    else {
        msgBox.setText(«Error saving.");
        msgBox.setWindowTitle("Error!");
    } msgBox.exec();
}
```

Чтобы начертить график функции в дополнительной секции, нужно его выбрать из списка и задать значения. Перед каждой строчкой нужно нажать кнопку «Set», чтобы значения передать переменным.

Результаты исследования и их обсуждение

Разработанный программный ресурс позволяет решать две важнейшие задачи. Во-первых, он может использоваться на уроках математики для объяснения нового материала по теме «Графики математических функций», а во-вторых – для ознакомления с возможностями языка программирования C++.

Язык C++ может использоваться не только для начального изучения основ программирования, но и для разработки программных продуктов. Однако для разработки графического интерфейса необходимо использовать дополнительные средства. В нашем случае разработка осуществлялась с помощью программы *QtDesigner*. Это достаточно сложный и трудоемкий процесс, который может занять больше времени, чем разработка продуктов с помощью компонентов Microsoft Visual Studio. Разработанная виртуальная лаборатория по математике имеет достаточно большой вес. Кроме того, при тестировании работы программы на ОС Windows 10 были выявлены проблемы совместимости.

Отметим, что возможности виртуальной лаборатории были продемонстрированы учащимся 8-го класса во время объяснения материала по теме «Основные алгоритмические конструкции». Учащиеся интересовались возможностями языка программирования C++ и историей его создания, местом C++ в рейтинге языков программирования,

достоинствами и недостатками C++ по сравнению с C#. Отдельному обсуждению был посвящен вопрос о выборе языка программирования для решения задач на Едином государственном экзамене.

В дальнейшем планируется добавить в данный ресурс вкладку «Исследование функции» и протестировать его во время проведения интегрированного урока по математике и информатике в 10 классе.

Выводы

Электронные образовательные ресурсы могут выполнять различные функции в учебном процессе. С их помощью учитель может объяснять новый материал, контролировать ход выполнения заданий учащимися, определять их уровень знаний и умений. Наибольшую методическую ценность имеют ЭОР, разработанные в программных средствах или посредством языков программирования, которые изучают учащиеся. В этом случае они являются непосредственной демонстрацией возможностей данных средств и направлены на развитие познавательного интереса к их изучению.

Список литературы

1. Бужинская Н.В. Игровые методы в подготовке IT-специалистов // Наука и перспективы. 2018. № 1. С. 8–14.
2. Андрианова А.А. Алгоритмизация и программирование. Практикум: учеб. пособие. СПб: Лань, 2019. 240 с.
3. Рудаков А.В. Технология разработки программных продуктов: учебное пособие для студентов среднего профессионального образования. М.: Издательский центр «Академия», 2007. 208 с.
4. Шлее М.Ш. Qt 5.10. Профессиональное программирование на C++. СПб.: БХВ-Петербург, 2018. 1072 с.
5. Qt Designer Manual [Электронный ресурс]. URL: <https://doc.qt.io/qt-5/qt designer-manual.html> (дата обращения: 10.04.2019).