

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Методические указания
по выполнению лабораторных работ по дисциплине
«ОСНОВЫ КРОССПЛАТФОРМЕННОГО ПРОГРАММИРОВАНИЯ»
для студентов специальности 10.05.01 Компьютерная
безопасность
специализация «Информационно-аналитическая и техническая
экспертиза компьютерных систем»

Ставрополь
2026

Содержание

ВВЕДЕНИЕ	3
Лабораторная работа № 1	4
Лабораторная работа № 2	6
Лабораторная работа № 3	7
Лабораторная работа № 4	8
Лабораторная работа № 5	9
Лабораторная работа № 6	11
Лабораторная работа № 7	12
Лабораторная работа № 8	14
Лабораторная работа № 9	16

ВВЕДЕНИЕ

Дисциплина «Основы кроссплатформенного программирования» является одной из основных при подготовке специалистов в области компьютерной безопасности. Материал, представленный в учебном пособии по дисциплине «Основы кроссплатформенного программирования» направлен на формирование универсальных компетенций студентов специальности 10.05.01 «Компьютерная безопасность», специализации «Информационно-аналитическая и техническая экспертиза компьютерных систем». В том числе:

УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач знаниями для создания профессиональных программных продуктов.

Задачи учебного пособия (лабораторного практикума) совпадают с задачами изучения дисциплины «Основы кроссплатформенного программирования» и могут быть сформулированы следующим образом:

- применение современных методов и технологий программирования;
- порядок разработки современного кроссплатформенного программного обеспечения (ПО);
- современных тенденций разработки и сопровождения программного обеспечения;
- основных программных алгоритмов на языке C++ в среде Qt.

Учебное пособие по дисциплине «Основы кроссплатформенного программирования» представляет собой лабораторный практикум по дисциплине «Основы кроссплатформенного программирования» и составлено в соответствии с рабочей программой дисциплины для студентов специальности 10.05.01 «Компьютерная безопасность», специализации «Информационно-аналитическая и техническая экспертиза компьютерных систем».

В методических указаниях приведены алгоритмы, структуры и технологии, широко используемые для разработки современного прикладного программного обеспечения. Структурно каждая из работ, в предлагаемом учебном пособии, состоит из краткого изложения теоретического материала, необходимого для ее выполнения, листингов с примерами исходного кода на языках C и C++, заданий к лабораторной работе, методики выполнения заданий, контрольных вопросов, списка литературы, рекомендуемой к использованию в данной теме.

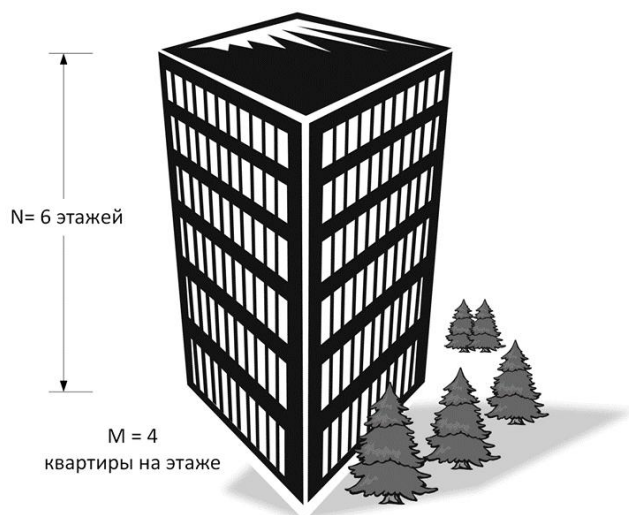
Лабораторная работа № 1

Первая программа Qt

Ваша консольная программа должна обладать интерактивным интерфейсом, для этого вспомните каким образом в языке Си осуществляются операции ввода-вывода.

Задание 1

Дано: дом с одним подъездом высотой в 6 этажей, на лестничной площадке по 4 квартиры.



Напишите программу для лифта, который после нажатия пассажиром кнопки выбора этажа высвечивает номера квартир на данном этаже, например так

Этаж 1 → 1 – 4

Этаж 2 → 5 – 8

Задание 2

Дано: решите задачу обратную заданию 1. На этот раз ваша программа должна подсказывать пользователю на каком этаже расположена та или иная квартира, например так

Кв 1 → 1 этаж

Кв 8 → 2 этаж

Задание 3

Дано: N-этажный дом с K-подъездами, на лестничной площадке по M квартир. Ваша программа должна подсказывать пользователю в каком подъезде и на каком этаже расположена та или иная квартира.

Задание по вариантам

Напишите на языке Си программы осуществляющие следующие расчеты:

Вариант	Задание 1	Задание 2	Задание 3
1	$\frac{x^2 + 3x + y}{ax + e^y}$	$\frac{5,23 + 7,6^x + \sin \frac{\pi}{4}}{\sin \frac{2\pi}{7} + 3,1}$	$2^{-x} \sqrt{x + \sqrt[4]{ x }}$
2	$\frac{c + d - x}{ x + d}$	$6,673 \times 10^{-8} \frac{m_1 m_2}{r^2}$	$\sqrt{ x - 1 + \sin(x)}$
3	$\sqrt[8]{x^8 + 8^x}$	$1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!}$	$\frac{xyz - 3,3 x + \sqrt[4]{y} }{10^7 + \sqrt{lg4!}}$
4	$\frac{xyz - 3,3 x + \sqrt[4]{y} }{10^7 + \sqrt{lg4!}}$	$x \cos x + \sin^3 x$	$6,673 \times 10^{-8} \frac{m_1 m_2}{r^2}$
5	$\frac{\beta + \sin^2 \pi^4}{\cos 2 + ctgy }$	$1 + \frac{1}{x} + \frac{1}{x^2}$	$\frac{c + d - x}{ x + d}$
6	$1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!}$	$\sqrt[8]{x^8 + 8^x}$	$\frac{5,23 + 7,6^x + \sin \frac{\pi}{4}}{\sin \frac{2\pi}{7} + 3,1}$
7	$6,673 \times 10^{-8} \frac{m_1 m_2}{r^2}$	$2^{-x} \sqrt{x + \sqrt[4]{ x }}$	$\frac{\beta + \sin^2 \pi^4}{\cos 2 + ctgy }$
8	$e^{ x-y } + \ln(1 + e) \log_2 tg 2$	$\frac{c + d - x}{ x + d}$	$\sqrt[8]{x^8 + 8^x}$
9	$\sqrt{ x - 1 + \sin(x)}$	$\frac{xyz - 3,3 x + \sqrt[4]{y} }{10^7 + \sqrt{lg4!}}$	$e^{ x-y } + \ln(1 + e) \log_2 tg 2$
10	$2^{-x} \sqrt{x + \sqrt[4]{ x }}$	$\frac{\beta + \sin^2 \pi^4}{\cos 2 + ctgy }$	$1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!}$

Ваша программа должна предусматривать ввод пользователем исходных данных и вывод результата расчетов на экран.

Лабораторная работа № 2

Условные операторы и циклы

1. Объявите одномерный массив типа данных T (см. таблицу) размерности N.
2. Заполните массив случайными значениями в диапазоне от MIN до MAX.
3. Выведите массив на экран.
4. Используя цикл for найдите сумму элементов массива.
5. Используя цикл while..do найдите среднее арифметическое значение из хранимых в массиве значений.
6. Используя цикл do..while найдите в массиве самое большое и самое маленькое значение.
7. Объявите 2-хмерный массив типа данных T размерности NxN.
8. Используя любой цикл, заполните 2-хмерный массив следующим образом:
 - a. В главную диагональ поместите значение $(MIN+MAX)/2$;
 - b. Ниже главной диагонали поместите значение MIN
 - c. Выше главной диагонали поместите значение MAX

Вариант	Тип данных T	Размерность N	MIN-MAX
1	T = short int	50	-20...20
2	T = int	20	-1000...1000
3	T = float	20	-10,00...10,00
4	T = double	20	-1,00...1,00
5	T = unsigned short int	255	10...90
6	T = unsigned int	50	0..100
7	T = short int	50	-25...25
8	T = unsigned short int	25	0..50
9	T = long int	25	-100..100
10	T = unsigned long int	25	0..200

Общее задание:

Напишите консольное приложение-калькулятор, которое позволит пользователю осуществлять следующие операции:

1. Сложение, вычитание, деление, умножение.
2. Возведение в квадрат.
3. Взятие квадратного корня.
4. Инвертировать введенное число (менять знак на противоположный).
5. Возвращать число пи

Лабораторная работа № 3

Функции

В качестве исходных данных воспользуйтесь одномерным массивом типа данных T размерности N (см. задание своего варианта в ЛР №2). Напишите следующие функции:

1. Функция вывода содержимого массива на экран.
2. Функция, возвращающая сумму элементов массива.
3. Функция, возвращающая среднее арифметическое элементов массива.
4. Функция, замещающая значения всех элементов массива на значение типа данных T, передаваемое через аргумент функции.
5. Функция, изменяющая порядок следования элементов массива на обратный (элемент с индексом 0 попадает в ячейку N-1, 2 в N-2, и т.д.).
6. Разработайте шаблон функции, работающей с массивом любого типа данных. Задача функции – найти и вернуть индекс и значение минимального элемента в массиве.
7. Разработайте шаблон функции, работающей с массивом любого типа данных. Задача функции – любым известным вам способом упорядочить элементы массива по возрастанию.
8. Напишите функцию на вход которой подается произвольный массив символов unsigned char, функция должна преобразовать символы в их числовой код и вернуть сумму этих чисел.

Доступ к массиву должен осуществляться только через параметры функции.

Лабораторная работа № 4

Объектно-ориентированная модель Qt

Разработайте следующие классы:

Вариант 1.

1. Класс “Воздушный шар”. Умения класса: взлет и посадка.
2. Класс “Дирижабль” и класс “Двигатель”. Умения класса “Дирижабль”: взлет, посадка, движение со скоростью 0-100 км.ч. Умения класса “Двигатель”: вкл. и откл. Класс “Дирижабль” должен уметь отправлять сигналы “Взлет” и “Посадка”.
3. Класс “Диспетчер”. Обладает слотом способным получать сигналы от класса “Дирижабль”.

Вариант 2.

1. Класс “Батискаф”. Умения класса: открыть люк для посадки экипажа, закрыть люк, погружение на глубину до 3000 м. Всплытие. Батискаф должен уметь отправлять сигналы об открытии и закрытии люка.
2. Класс “Подводная лодка” и класс “Двигатель”. Умения класса “Подводная лодка”. Открыть люк, закрыть люк, погружение на глубину до 300 м. Всплытие. Движение под водой со скоростью 0-10 узлов в час, движение над водой 0-25 узлов в час. Умения класса “Двигатель”: вкл. и откл.
3. Класс “Капитанский мостик”. Обладает слотом, способным получать сигналы от закрытии и открытии люка.

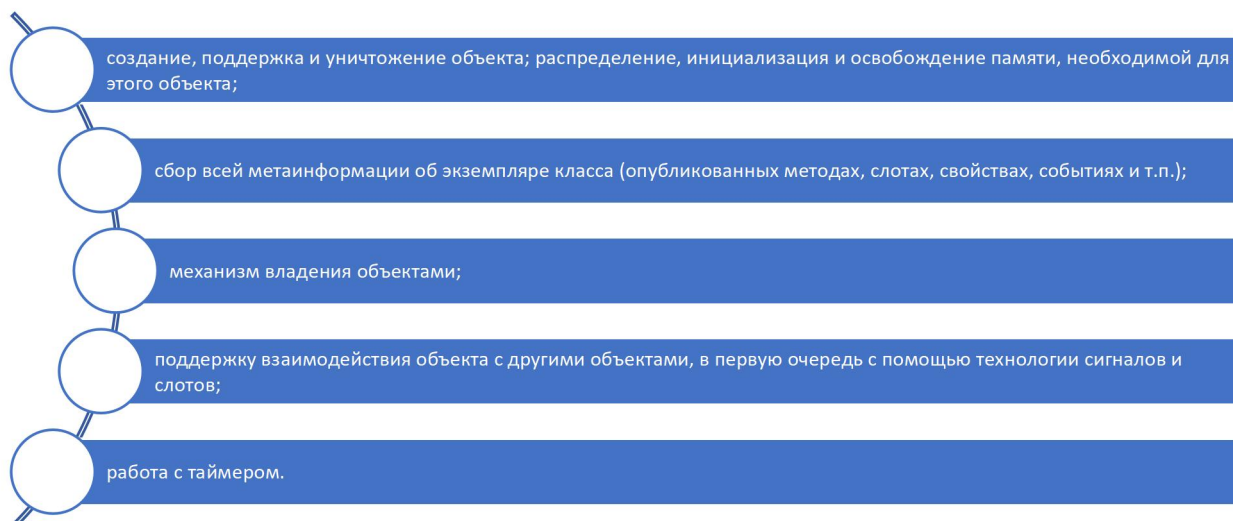
Вариант 3.

1. Класс “Чайник”. Умения класса: залить воду 0-2 литра, вылить воду.
 2. Класс “Электрочайник” и класс “Нагреватель”. Умения класса “Электрочайник”. Залить воду 0-2 литра, вылить воду. Нагреть воду до температуры 60-100 градусов. Умения класса “Нагреватель”: вкл. и откл. Класс “Электрочайник” должен уметь отправлять сигнал о том, что он включился и выключился.
 3. Класс “Наблюдатель”. Обладает слотом способным получать сигналы от класса “Электрочайник”.
-

Лабораторная работа № 5

QObject и QWidget

Большинство классов объектно-ориентированной библиотеки Qt берут свое начало от базового класса QObject. Исключение составляет лишь немногие классы, например шаблоны из библиотеки STL (Standard Template Library) или классы событий. Своих многочисленных потомков класс QObject наделяет целой плеядой важнейших для Qt возможностей.



```
1  #include "widget.h"
2
3  #include <QApplication>
4
5  int main(int argc, char *argv[])
6  {
7      QApplication a(argc, argv);
8      Widget w;
9      w.show();
10     return a.exec();
11 }
12
```

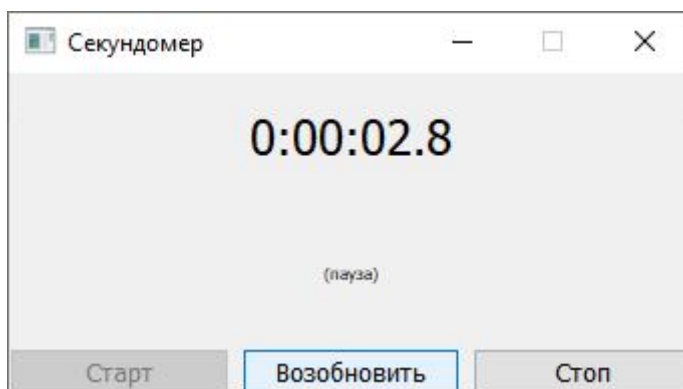
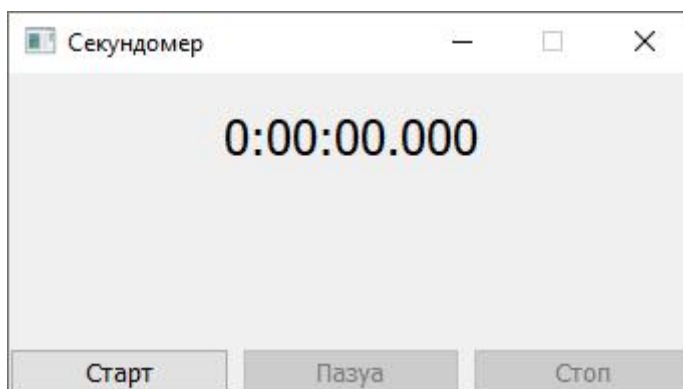
```
2
3 Widget::Widget(QWidget *parent)
4     : QWidget(parent)
5 {
6     label1=new QLabel("Метка", this);
7     label1->setGeometry(10,10,200,30);
8     label1->setFrameStyle(QFrame::Box | QFrame::Raised);
9
10
11     lineEdit1 = new QLineEdit(this);
12         lineEdit1 ->setGeometry(10,60,200,30);
13         lineEdit1 ->setPlaceholderText("Введите текст");
14
15     button1=new QPushButton("Отправить текст в метку",this);
16         button1->setGeometry(10,90,200,30);
17
18     connect(button1,SIGNAL(clicked()),this,SLOT(slot_click()));
19     connect(this,&Widget::signal_sendText,label1,&QLabel::setText);
20
21
22     label2=new QLabel(this);
23     label2->setGeometry(220,10,200,60);
24     label2->setWordWrap(true);
25     label2->setText("Пример передачи текста из строки ввода в метку через параметр сигнала");
26 }
27
28 void Widget::slot_click()
29 {
30     sendText(lineEdit1->text());
31 }
32
33 void Widget::sendText(const QString s)
34 {
35     emit signal_sendText(s);
36 }
37
```

Лабораторная работа № 6

События виджета

Разработайте приложение “Секундомер”.

1. Приложение должно позволять отсчитывать интервал времени с точностью до 100 миллисекунд, приостанавливать и возобновлять работу таймера.
2. Все операции должны выполняться с помощью команд QAction.
3. При проектировании интерфейса секундомера воспользуйтесь менеджерами компоновки.



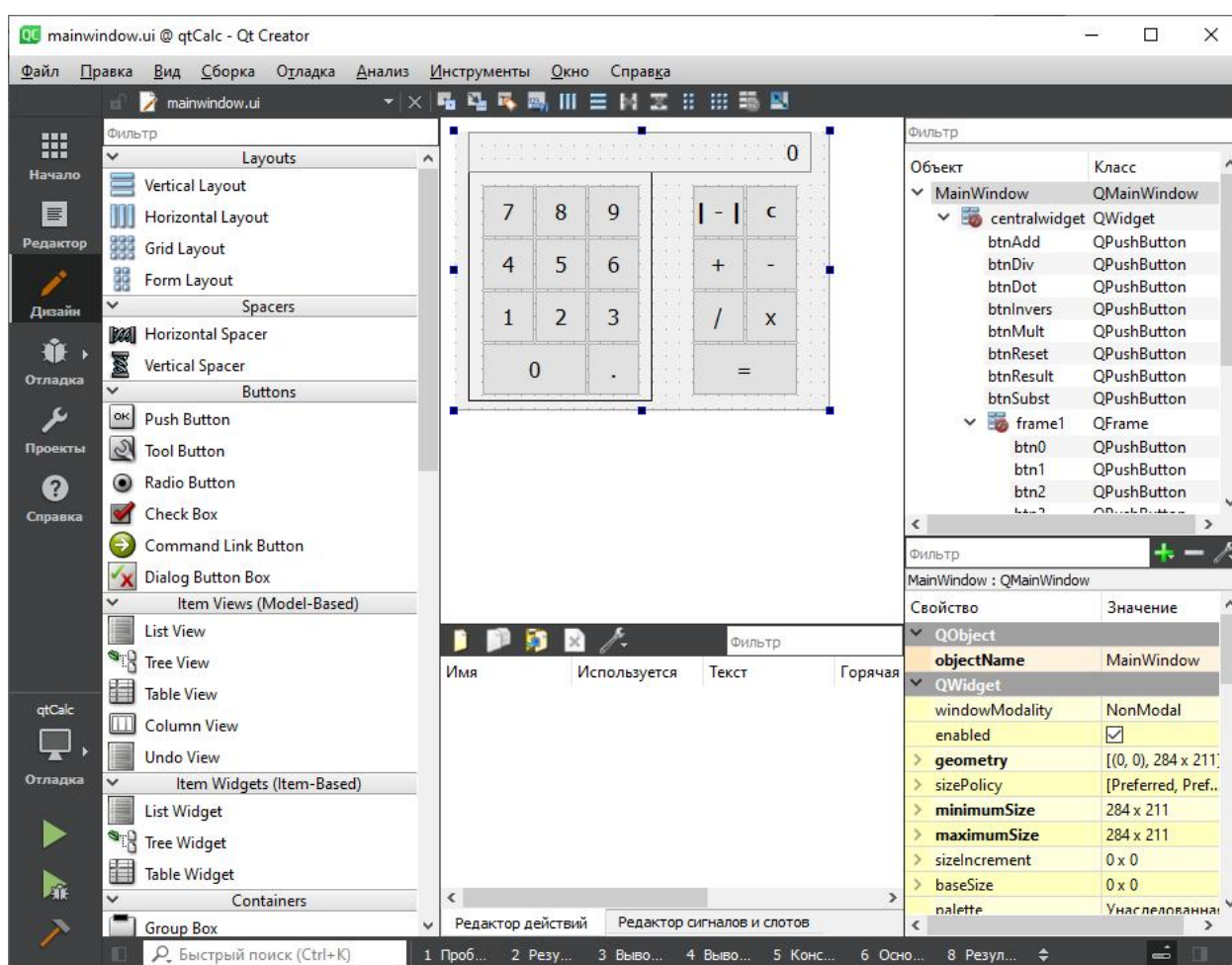
Лабораторная работа № 7

Визуальное проектирование

1. Изучите назначение следующих виджетов:

- a. QPushButton
- b. QRadioButton
- c. QCheckBox
- d. QListView
- e. QGroupBox
- f. QLabel
- g. QLineEdit
- h. QTextEdit

Какие методы, сигналы и слоты этих виджетов на ваш взгляд являются основными и определяют поведение этих элементов управления. Отметьте это в отчете.



2. Разработайте приложение “Калькулятор”

Вам понадобятся следующие классы:

1. Кнопка QPushButton.
2. Фрейм QFrame упрощает группировку элементов управления.
3. Метка QLabel позволит отобразить результаты расчетов.

Советы:

1. В момент щелчка по кнопке она генерирует сигнал clicked().

-
2. Для реакции на сигналы кнопок вам придется разработать соответствующие слоты и подключить их к кнопкам.
 3. Для того, чтобы поместить в метку QLabel текст вам пригодиться метод setText().

Задание к лабораторной работе № 6

1. Воспользуйтесь клавиатурным событием QKeyEvent и доработайте калькулятор (лаб. раб. №5) таким образом, чтобы пользователь смог осуществлять вычисления (ввод значений, выбор арифметических операций, и т.п.) с помощью клавиатуры.
2. Воспользуйтесь событием мышки QMouseEvent и доработайте калькулятор таким образом, чтобы пользователь смог, передвигая мышкой виджеты самостоятельно настраивать графический интерфейс приложения под свои предпочтения.
3. **(Повышенная сложность)** Научите приложение калькулятора сохранять местоположение метки и виджетов-кнопок (см. п.2) во внешний файл при закрытии программы и загружать эти данные из файла при запуске программы.

Совет

При выполнении задания повышенной сложности, возможно, вам пригодится помощь класса QFile (см. пример в архиве exQFile + самостоятельное изучение класса).

Лабораторная работа № 8

Основные элементы управления

Благодаря `QAbstractButton` любой виджет-наследник может быть снабжен пояснительной надписью

```
void setText(const QString &text)
```

Для поддержки интуитивной понятности назначения кнопки на ней разрешено разместить небольшую картинку в формате `QIcon`

```
void setIcon(const QIcon &icon)
```

Для ускоренного доступа к функционалу кнопки без применения мышки возможно задать комбинацию быстрых клавиш

```
void setShortcut(const QKeySequence &key)
```

По умолчанию щелчок мышкой по кнопке `QPushButton` и `QToolButton` приводит к тому, что кнопка визуально утапливается и сразу же возвращается в исходное не нажатое состояние. Такое поведение соответствует режиму работы кнопок `isChecked() == false`. Если воспользоваться методом

```
void setCheckable(bool)
```

отправив в него `true`, то мы сможем первым щелчком фиксировать кнопку в нажатом состоянии, а вторым – снимать фиксацию и возвращать кнопку в исходное. Для проверки не зафиксирована ли кнопка воспользуйтесь методом

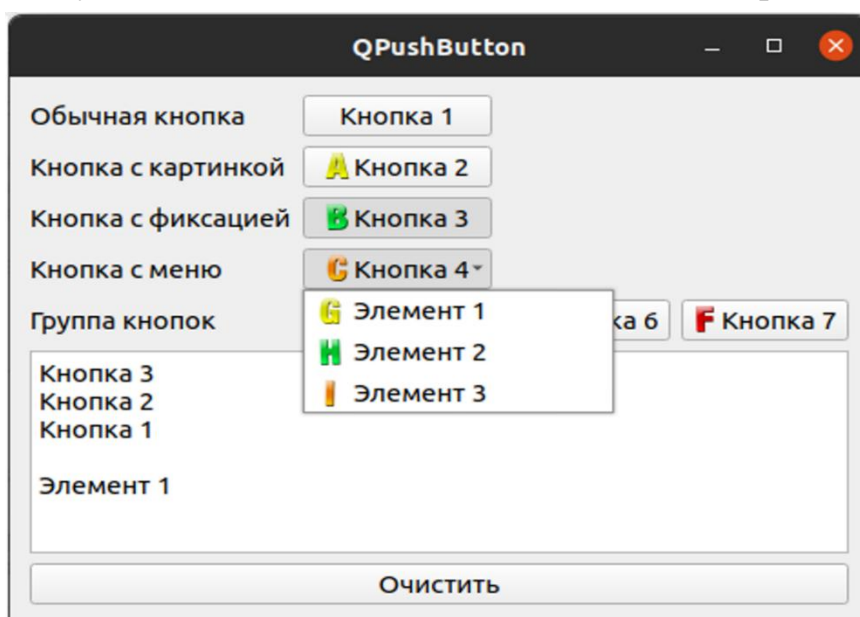
```
bool isDown() const
```

Для фиксации и снятия с фиксации кнопки из кода программы обратитесь к `setDown()`.

Если кнопка-переключатель `QRadioButton` или `QCheckBox` отмечена “галочкой” об этом расскажет метод

```
bool isChecked() const
```

Для установки или снятия отметки с кнопки обратитесь к `setChecked()`.



Кнопка-флажок `QCheckBox`

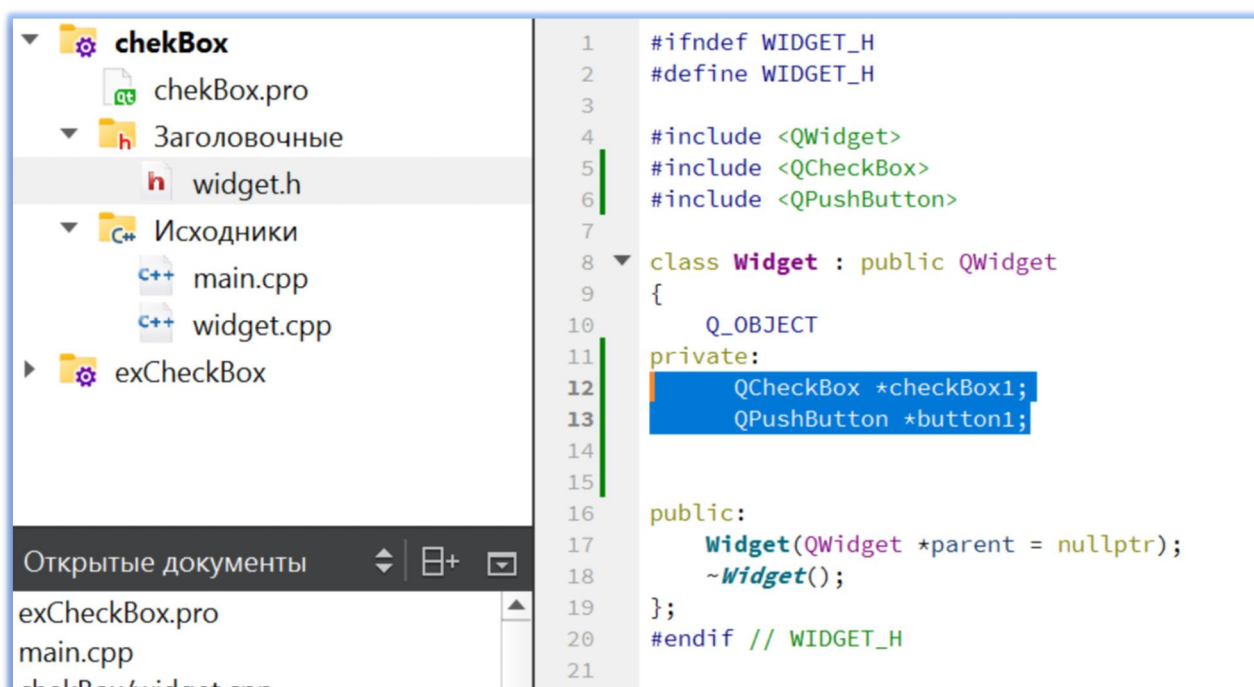
По умолчанию виджет работает в двухзначной логике, т.е. неопределенное состояние `Qt::PartiallyChecked` у флажка отключено, но если вам понадобится все три состояния обратитесь к методу

```
void QCheckBox::setTristate(bool y = true)
```

Аргумент `true` переведет виджет в трехзначную логику, разрешив состояние `Qt::PartiallyChecked`, `false` – возвратит виджет в исходную двухзначную логику.

Для изменения состояния кнопки из кода программы можно воспользоваться методом

```
void QCheckBox::setCheckState(Qt::CheckState state)
```

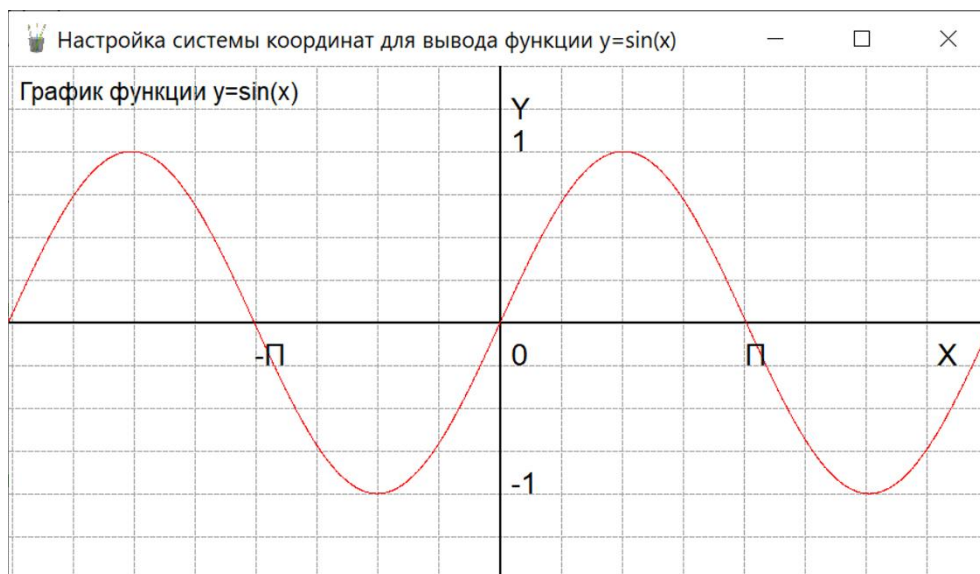


Лабораторная работа № 9

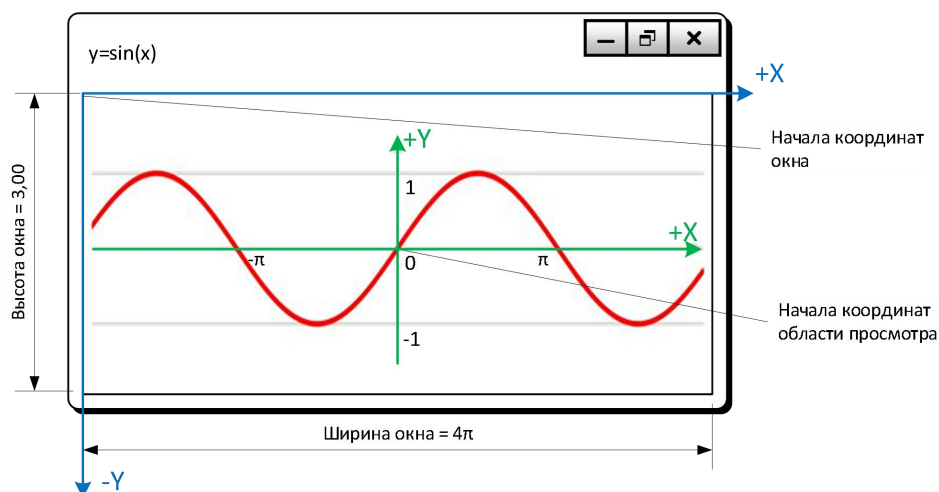
Графика в Qt

Напишите программу позволяющую рассчитывать и выводить на экране графики функций:

1. $\sin()$
2. $\cos()$
3. $\text{Tg}()$
4. И т.п.
- 5.



Идея: Мы с вами уже привыкли работать в логической системе координат Qt с точкой отсчета (0, 0) начинающейся в левом верхнем углу виджета и координатными осями, направленными сверху вниз и слева-направо. В этой системе координат в качестве единицы измерения принят экранный пиксель, а текстовые надписи выводятся слева-направо. А что, если такое поведение координатной системы далеко не догма и его относительно просто изменить?



Подсказка: для выполнения лабораторной работы вам возможно понадобится изучить методы управления логическим окном и областью просмотра:

```
void QPainter::setWindow(int x, int y, int width, int height)
```

```
void QPainter::setViewport(int x, int y, int width, int height)
```

1. Павловская, Т. А. C/C++ : программирование на языке высокого уровня : учебник для вузов / Т. А. Павловская. - СПб. [и др.] : Питер, 2010. - 461 с. - (Учебник для вузов). - Библиогр.: с. 383. - Алф. указ.: с. 450-460. - Прил.: с. 384-449. - ISBN 978-5-94723-568-5
2. Страуструп, Б.
Язык программирования С++ для профессионалов
Электронный ресурс : учебное пособие / Б. Страуструп. - Язык программирования С++ для профессионалов, 2021-01-23. - Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. - 670 с. - Книга находится в базовой версии ЭБС IPRbooks.