

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра информационных систем

ПРАКТИЧЕСКАЯ РАБОТА №8
по дисциплине «Объектно-ориентированное программирование»
ТЕМА: ПОСТРОЕНИЕ ГРАФИКОВ

Студенты гр. 3372

Преподаватель

Беляев К.В.

Лазарев Ф.Н.

Егоров С.С.

Санкт-Петербург

2024

Задание на Практическую работу

Студенты Беляев К.В., Лазарев Ф.Н.

Группа 3372

Исходные данные: Разработать GUI приложение на основе работы №7 для построения графиков функций синус и интегральный синус на множестве вещественных чисел.

Спецификации классов

Таблица 1. Протокол класса Array

Атрибуты(старые)

идентификатор	тип	область видимости	семантическое описание
length	int	private	Целочисленная длина массива
arr	number*	private	Указатель на первый элемент массива
Методы(старые)			
идентификатор	область видимости		семантическое описание
Array	public		Конструктор класса. Создает массив заданной длины, по умолчанию – 0
~Array	public		Деструктор класса
getLength	public		Получение длины массива
fill	public		Заполнение массива числами с консоли
resize	public		Изменение размера массива
changeElement	public		Изменение выбранного элемента числом с консоли
printArray	public		Вывод массива в консоль
averageValue	public		Подсчет среднего значения элементов массива
SKO	public		Подсчет СКО элементов массива
shakerSort	public		Сортировка массива по убыванию – если передается параметр 1, по возрастанию – если передается 0 или не передается ничего

Таблица 2. Протокол класса TComplex

Атрибуты(старые)			
идентификатор	тип	область видимост и	семантическое описание
re	double	private	Вещественная часть комплексного числа
im	double	private	Мнимая часть комплексного числа
separator	Qchar	private	Разделение аргументов в запросе
Методы(старые)			
идентификатор	область видимости		семантическое описание
TComplex()	public		Конструктор класса по умолчанию

TComplex(double re, double im)	public	Конструктора класса, принимающий вещественное и мнимое части комплексного числа
TComplex(double re)	public	Конструктор класса, принимающий вещественную часть комплексного числа
getRe	public	Получение вещественной части комплексного числа
getIm	public	Получение мнимой части комплексного числа
module	public	Вычисление модуля комплексного числа
operator+	public	Оператор сложения
operator-	public	Оператор вычитания
operator/	public	Оператор деления
operator*	public	Оператор умножения
operator+=		Оператор сложения с присваиванием
operator-=	public	Оператор вычитания с присваиванием
operator/=	public	Оператор деления с присваиванием
operator*=	public	Оператор умножения с присваиванием
operator=	public	Оператор присваивания
operator==	public	Оператор «равно»
operator!=	public	Оператор «неравно»
operator<	public	Оператор «меньше»
operator>	public	Оператор «больше»
pow	public	Вычисление корня из комплексного числа
setSeparator	public	Установление разделителя аргументов
Operator QString()	public	Преобразование в QString
operator >>	public	Оператор ">>" для работы с потоком ввода
operator <<	public	Оператор "<<" для работы с потоком вывода
operator >>	public	Оператор ">>" для работы с QByteArray

Таблица 3. Протокол класса Polynom

Атрибуты(старые)			
идентификатор	тип	область видимост и	семантическое описание
roots	number*	private	Массив корней полинома
coefficients	number*	protected	Массив коэффициентов полинами
degree	int	protected	Степень полинома
Методы(старые)			
идентификатор	область видимости		семантическое описание
Polynom()	public		Конструктор класса по умолчанию
polynomWithDegrees	public		Вывод полинома в каноническом виде
polynomWithRoots	public		Вывод полинома с корнями
valueAtPoint	public		высчитать значение в точке

Таблица 4. Протокол класса Tinterface

Атрибуты(старые)			
идентификатор	тип	область видимост и	семантическое описание
trigonometry	QLabel *	private	Лэйбл коэффициента
SinMode	QRadioButton *	private	выбор функции Sin
SiMode	QRadioButton *	private	выбор функции Si
semicolonIdentificator	QLabel *	private	лейбл “ ; ”
initialValueOfRangeLabel	QLabel *	private	?
finalValueOfRangeLabel	QLabel *	private	?
ReValueInitial	QLineEdit *	private	?
ReValueFinal	QLineEdit *	private	?
sinPlotter	SinPlotter *	private	Виджет для отображения графика Sin
siPlotter	SiPlotter *	private	Виджет для отображения графика Si

updatePlotButton	QPushButton *	private	Кнопка отображения графика
values	double *	private	Массив значений функции
Методы(старые)			
идентификатор	область видимости		семантическое описание
Tinterface	public		Конструктор класса
~Tinterface	public		Деструктор класса
updatePlot	public slot		Обновление графика

Таблица 5. Протокол класса TFunction

Методы(старые)		
идентификатор	область видимости	семантическое описание
TFunction()	public	Конструктор класса по умолчанию
value()	public	Вычисление значения функции в точке

Таблица 6. Протокол класса TFSin

Атрибуты(старые)			
идентификатор	тип	область видимости	семантическое описание
derivatives	number*	private	Массив производных
Методы(старые)			
идентификатор	область видимости		семантическое описание
TFSin()	public		Конструктор класса по умолчанию
derivativesCalc()	public		Вычисление производных
~TFSin()	public		Деструктор класса по умолчанию

Таблица 7. Протокол класса TFSin

Атрибуты(старые)

идентификатор	тип	область видимост и	семантическое описание
derivatives	number*	private	Массив производных
Методы(старые)			
идентификатор	область видимости		семантическое описание
TFSi()	public		Конструктор класса по умолчанию
derivativesCalc()	public		Вычисление производных
~TFSi()	public		Деструктор класса по умолчанию

Таблица 8. Первичный протокол класса SiPlotter

Атрибуты			
идентификатор	тип	область видимост и	семантическое описание
siValues	double*	private	массив значений функции
size	int	private	размер массива значений
x1	double	private	начало диапазона
x2	double	private	конец диапазона
Методы			
идентификатор	область видимости		семантическое описание
SiPlotter()	public		Конструктор класса по умолчанию
clearPlot	public		очищение графика
setSiValues	public		добавить в график точки функций
paintEvent	public		Отобразить график
~TFSi()	public		Деструктор класса по умолчанию

Таблица 9. Первичный протокол класса SinPlotter

Атрибуты

идентификатор	тип	область видимост и	семантическое описание
sinValues	double*	private	массив значений функции
size	int	private	размер массива значений
x1	double	private	начало диапазона
x2	double	private	конец диапазона
Методы			
идентификатор	область видимости		семантическое описание
SiPlotter()	public		Конструктор класса по умолчанию
clearPlot	public		очищение графика
setSinValues	public		добавить в график точки функций
paintEvent	public		Отобразить график
~TFSi()	public		Деструктор класса по умолчанию

Диаграмма классов

На рисунке 1 представлена диаграмма классов, дополненная атрибутами и методами.

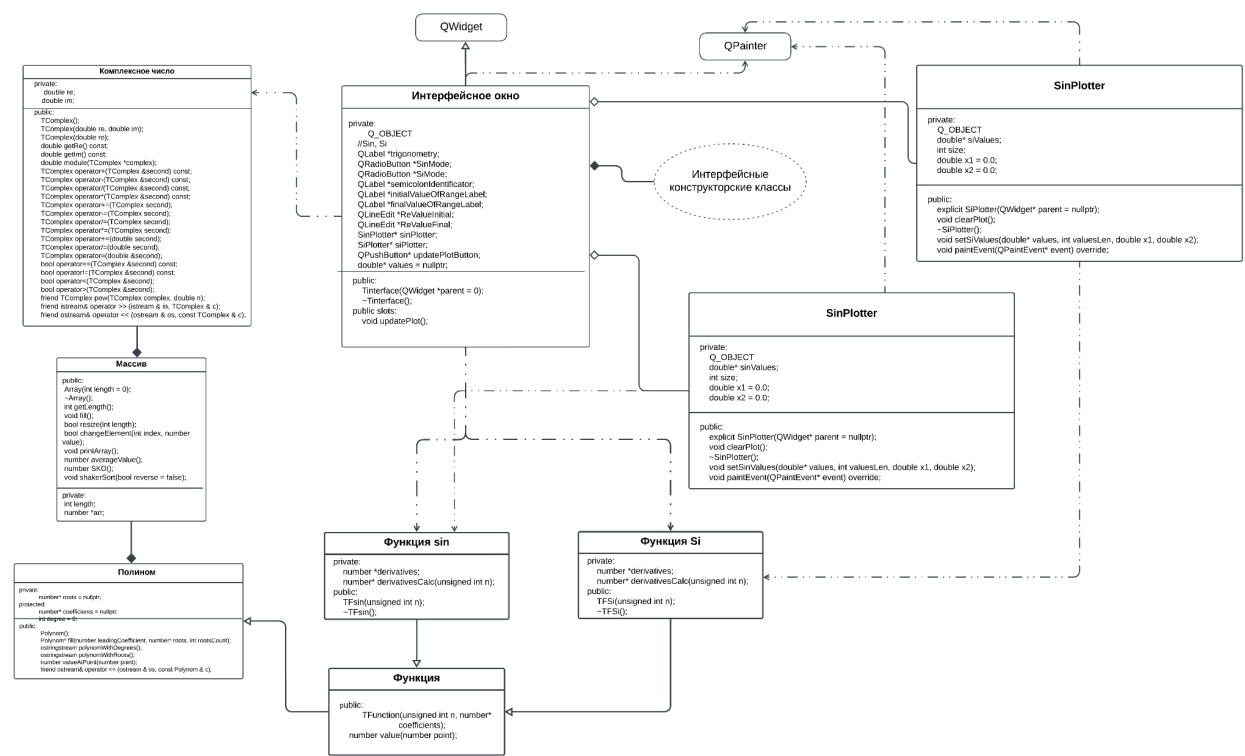


Рисунок 1 – Диаграмма классов.

Описание контрольного примера с исходными и ожидаемыми расчетными данными

1. Выбираем функцию для отображения графика: Sin
2. Вводим диапазон от -180° до 180°
3. Нажимаем кнопку “Обновить график”, и выводится график функции в заданном диапазоне.
4. Изменим функцию на Si и нажмем кнопку “Обновить график”

СКРИНШОТЫ РАБОТЫ ПРОГРАММЫ НА КОНТРОЛЬНЫХ ПРИМЕРАХ

После запуска программы на экране появляется окно, с кнопками, полями ввода и полями с текстом на рисунке 2.

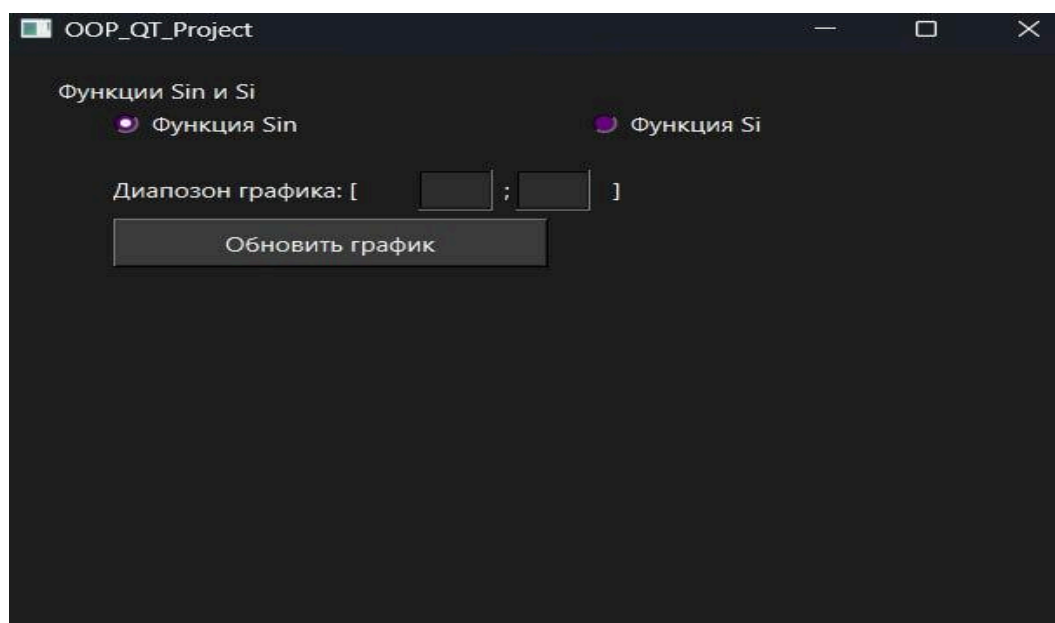


Рисунок 2 – Выбор функции из контрольного примера

Необходимо выбрать диапазон графика в градусах

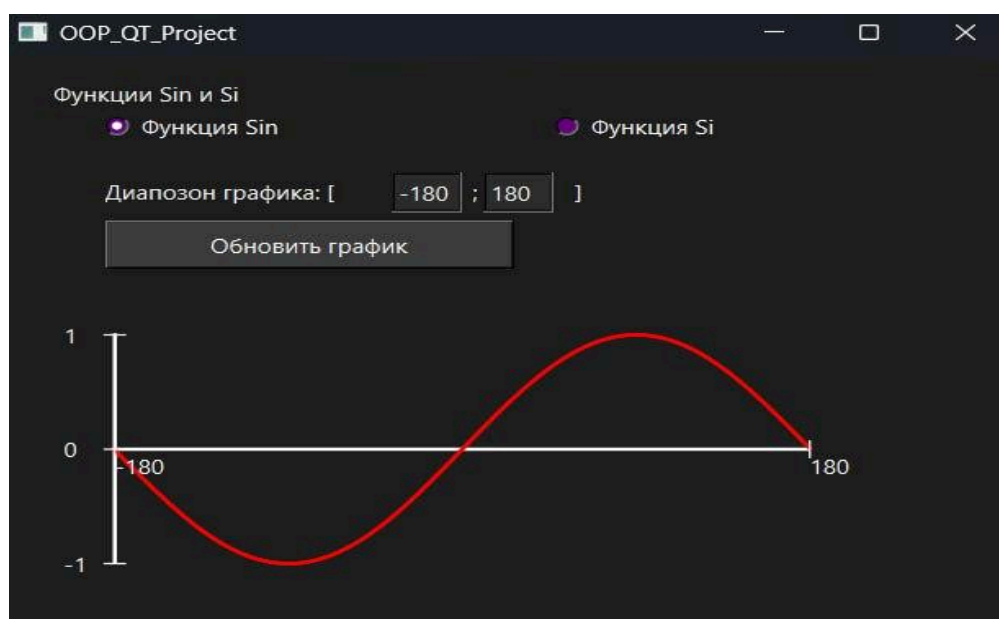


Рисунок 3 – Вывод графика

Изменим функцию на Si. На рисунке 5 показаны результаты работы программы.

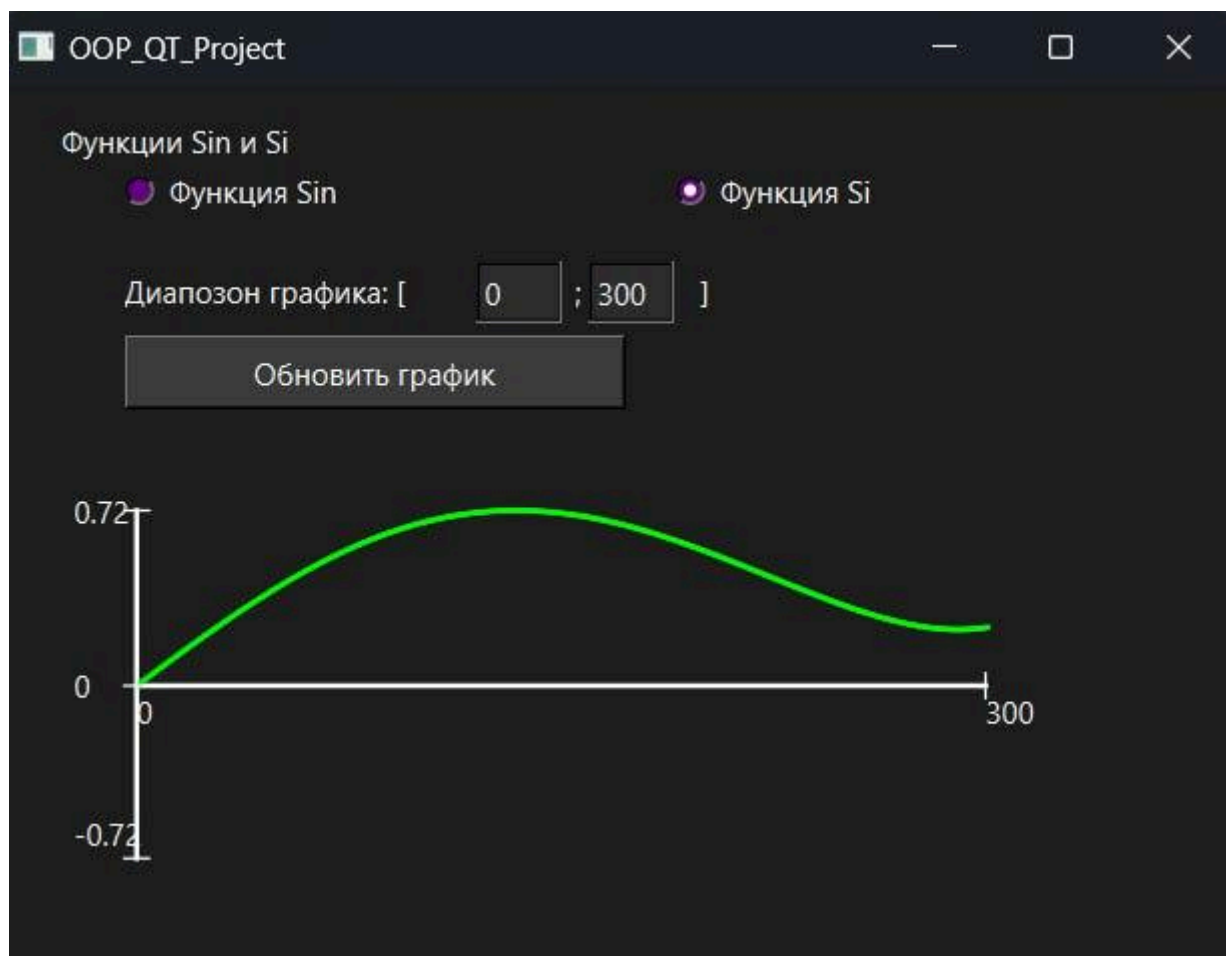


Рисунок 5 – Изменение графика функции

Наконец, чтобы выйти из программы, нужно нажать на крестик в правом верхнем углу, и программа автоматически закроется.

ВЫВОДЫ ПО ВЫПОЛНЕНИЮ РАБОТЫ

В ходе выполнения практической работы была реализована и отлажена программа, предназначенная для создания GUI приложения с клиентской и серверной частью, реализующего функции перечисленные в описании работы №7, с добавлением функционала рисования графика функции на заданном промежутке.