

Лабораторная работа 3.

Графический интерфейс пользователя (GUI).

GUI играет ключевую роль в восприятии и удобстве использования программного обеспечения. GUI позволяет пользователям взаимодействовать с компьютерными программами и устройствами более интуитивно, используя визуальные элементы, такие как окна, кнопки, меню и иконки, вместо текстовых команд. Разработка GUI-приложений требует глубокого понимания принципов дизайна интерфейса, а также умения работать с специализированными инструментами и библиотеками.

Язык программирования C++ обладает широким спектром доступных библиотек для разработки GUI:

Qt https://wiki.qt.io/Qt_for_Beginners

wxWidgets <https://www.wxwidgets.org/>

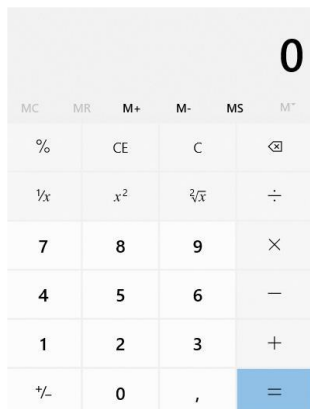
Window Forms <https://www.youtube.com/watch?v=IYLYX5Ei48I>

И другие... В этой работе нет ограничения по использованию какой-то библиотеки, пользуйтесь той которая кажется удобна вам, но будьте готовы ответить почему выбрали именно ее... Единственное что предполагается, что все задания должны быть выполнены с применением одной и той же библиотеки.

Целью данных заданий является не только освоение технических аспектов создания GUI на C++, но и развитие навыков планирования и проектирования пользовательского интерфейса, что является важной частью процесса разработки программного обеспечения.

Задание 1. Калькулятор на C++ (Максимум 8 баллов).

Вам необходимо создать калькулятор, аналог обычного калькулятора под Windows. Интерфейс может быть изменен, главное выполнение ключевых функций.



Данное задание имеет два уровня сложности:

1. Реализация простого калькулятора как на скриншоте выше, т.е. вам необходимо реализовать набор кнопок для ввода цифр и арифметических операций, интерфейс промежуточных вычислений и сам ответ (**максимум 3 балла**).

2. Калькулятор способный парсить математические выражения.

Парсер — это программа, анализирующая входное арифметическое выражение. Программы подобного класса, иногда называют так же «распознавателями» (**максимум 5 баллов**).

Парсинг — процесс разбора входного арифметического выражения на более простые составляющие.

Пример: <https://habr.com/ru/post/263775/>

- Должно быть реализовано окно, в которое можно ввести выражение, например, $\sin(x) * 5 * (5x + 7)$.
- Также окно для ввода значения переменной x .
- Результат вычисленного значения выражения.

Правила ввода выражений должны соответствовать стандартам. Данную задачу можно решить с использованием ООП классов или структур каждая из которых будет хранить информацию либо об операции, либо о значении переменной. Реализуйте на выбор 10 функций из представленного списка ниже (простые арифметические выражения не учитываются).

Оператор	Значение
-	вычитание
*	умножение
/	деление
n	возведение в степень: x^n - x в степени n
$^{(1/n)}$	корень степени n : $x^{(1/n)}$
()	скобки
	скобки модуля: $ x $

Список функций

Имя	Описание
<code>log2(x)</code>	логарифм по основанию 2 от x
<code>lg(x)</code> или <code>log10(x)</code>	логарифм по основанию 10 от x
<code>log(x;b)</code>	логарифм x по основанию b <code>log(x;3)</code>
<code>ln(x)</code>	натуральный логарифм (логарифм по основанию e (2.71828...)) от x
<code>exp(x)</code> или <code>e^x</code>	экспонента от x (e в степени x)
<code>sqrt(x)</code>	квадратный корень из x
<code>sign(x)</code>	функция знака: -1 если $x < 0$, 1 если $x > 0$ и 0 если $x = 0$

Встроенные константы

Имя	Описание
<code>pi</code>	Пи = 3,14...
<code>e</code>	e = 2,71828... число Эйлера

Тригонометрические функции

<code>sin(x)</code>	синус x
<code>cos(x)</code>	косинус x
<code>tg(x)</code> или <code>tan(x)</code>	тангенс x
<code>ctg(x)</code> или <code>cot(x)</code>	котангенс x
<code>arcsin(x)</code> или <code>asin(x)</code>	арксинус x
<code>arccos(x)</code> или <code>acos(x)</code>	арккосинус x
<code>arctg(x)</code> или <code>atan(x)</code>	арктангенс x
<code>arcctg(x)</code> или <code>acot(x)</code>	арккотангенс x
<code>sinh(x)</code> или <code>sh(x)</code>	гиперболический синус x
<code>cosh(x)</code> или <code>ch(x)</code>	гиперболический косинус x
<code>tanh(x)</code> или <code>th(x)</code>	гиперболический тангенс x
<code>coth(x)</code> или <code>cth(x)</code>	гиперболический котангенс x
<code>asinh(x)</code>	гиперболический арксинус x
<code>acosh(x)</code>	гиперболический арккосинус x
<code>atanh(x)</code>	гиперболический арктангенс x
<code>acoth(x)</code>	гиперболический арккотангенс x

Задание 2. Визуализация данных (Максимум 6 баллов).

В данной работе вам дан файл covid.csv, в котором находятся всего два столбца: дата и число новых случаев заболевания в этот день в России.

CSV (от англ. Comma-Separated Values — значения, разделённые запятыми) — текстовый формат, предназначенный для представления табличных данных. Строка таблицы соответствует строке текста, которая содержит одно или несколько полей, разделённых запятыми.

1. Считайте данные csv файла в любом удобном вам формате, продумайте эффективный способ хранения таких данных в памяти компьютера. Визуализируйте представленные данные, постройте график зависимости числа новых случаев заболевания от даты. По оси абсцисс откладываете дату, по оси ординат число новых случаев.

2. На вашей локальной машине разверните СУБД. Загрузите данные файла в таблицу, осуществите коннект с БД по средствам языка C++. Визуализируйте данные, аналогично заданию 1.

Продумайте дружелюбный интерфейс для пользователя.

Задание 3. Календарь событий (Максимум 21 балл)

Цель: Создать приложение на C++ с использованием библиотеки для GUI, которое позволяет пользователям просматривать, добавлять и удалять предстоящие события в календаре.

Обязательные Задачи:

- Разработать **класс Событие**, содержащий информацию о событии (например, название, дата и время, описание). (1 балл)
- Создать **GUI**, который отображает календарь, где пользователи могут выбирать дату и просматривать события на эту дату. (1 балл)
- Реализовать функционал **добавления и удаления событий**. (1 балл)
Дополнительные задачи*.
- **Асинхронная загрузка событий**. Когда пользователь выбирает дату в календаре, приложение может асинхронно загружать события для этой даты из файла или базы данных. Это позволит интерфейсу пользователя оставаться отзывчивым, пока выполняется загрузка данных (4 балла).
- **Уведомления о предстоящих событиях**. Реализуйте систему уведомлений, которая будет оповещать пользователя за определённое время до начала события. Это может быть всплывающее окно или звуковой сигнал (2 балла)
- **Повторение событий**. Добавьте возможность настройки повторяющихся событий: ежедневно, еженедельно, ежемесячно или

ежегодно. Это удобно для планирования регулярных встреч или мероприятий (2 балла)

- **Экспорт и импорт событий.** Разработайте функционал для экспорта и импорта событий из/в другие календарные приложения через стандартные форматы файлов, например, iCal или CSV (3 балла)
- **Поиск по событиям.** Реализуйте функцию поиска, позволяющую находить события по ключевым словам, датам, участникам или описанию (2 балла)
- **Адаптация интерфейса под пользователя.** Разработайте настройки интерфейса, позволяющие пользователю адаптировать внешний вид приложения под свои предпочтения (например, тёмный режим, размер шрифта). (2 балла)
- **Интеграция с облачными сервисами.** Обеспечьте возможность синхронизации событий с облачными сервисами (например, Google Calendar), чтобы пользователи могли получать свои события с любого устройства (3 балла)