БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Международный институт дистанционного образования

Кафедра «Информационные системы и технологии»

**КУРСОВОЙ ПРОЕКТ**

по учебной дисциплине

«СИСТЕМНОЕ ПРОГРАММИРОВАНИЕ»

Исполнитель: студент 3 курса, группы 41703120 Реут В.Л.

Руководитель: Радкевич А. С.

Минск 2023

**ОГЛАВЛЕНИЕ**

[ВВЕДЕНИЕ 3](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733098)

[ЗАДАНИЕ 1 4](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733099)

[Постановка задачи 4](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733100)

[Исходный код программы 4](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733101)

[ЗАДАНИЕ 2 9](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733102)

[Постановка задачи 9](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733103)

[***Диета.*** Определить иерархию овощей. Собрать в салат. Посчитать каллорийность. 9](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733104)

[Исходный код программы 9](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733105)

[Исходный код программы 16](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733106)

[Результат выполнения программы 17](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733107)

[ЗАДАНИЕ 3 18](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733108)

[Постановка задачи 18](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733109)

[Исходный код программы 18](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733110)

[Результат выполнения программы 24](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733111)

[ЗАКЛЮЧЕНИЕ 27](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733112)

[ЛИТЕРАТУРА 28](file:///D:\BNTU\BNTU\OOP\Course_Project_OOP.docx#_Toc125733113)

# **ВВЕДЕНИЕ**

Язык программирования С++ представляет высокоуровневый компилируемый язык программирования общего назначения со статической типизацией, который подходит для создания самых различных приложений. На сегодняшний день С++ является одним из самых популярных и распространенных языков.

С++ является мощным языком, унаследовав от Си богатые возможности по работе с памятью. Поэтому нередко С++ находит свое применение в системном программировании, в частности, при создании операционных систем, драйверов, различных утилит, антивирусов и т.д. К слову сказать, ОС Windows большей частью написана на С++

С++ является компилируемым языком, а это значит, что компилятор транслирует исходный код на С++ в исполняемый файл, который содержит набор машинных инструкций. Но разные платформы имеют свои особенности, поэтому скомпилированные программы нельзя просто перенести с одной платформы на другую и там уже запустить. Однако на уровне исходного кода программы на С++ по большей степени обладают переносимостью, если не используются какие-то специфичные для текущей ос функции. А наличие компиляторов, библиотек и инструментов разработки почти под все распространенные платформы позволяет компилировать один и тот же исходный код на С++ в приложения под эти платформы.

В 1979-80 годах Бьерн Страуструп разработал расширение к языку Си - "Си с классами". В 1983 язык был переименован в С++.

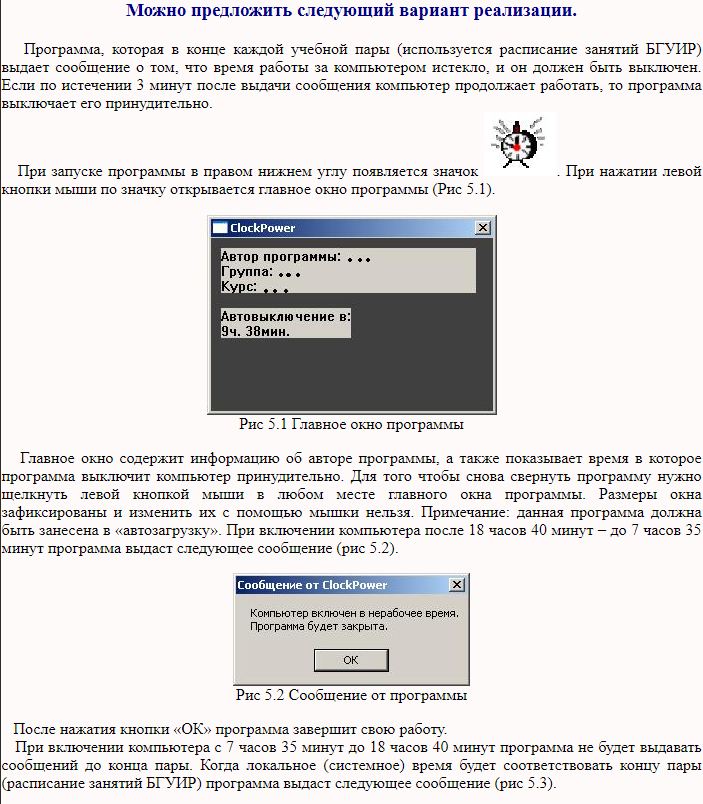
В 1985 году была выпущена первая коммерческая версия языка С++, а также первое издание книги "Языка программирования C++", которая представляла первое описание этого языка при отсутствии официального стандарта.

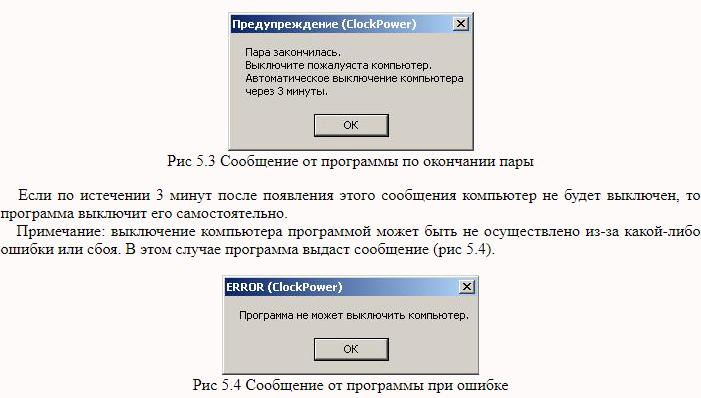
В 1989 была выпущена новая версия языка C++ 2.0, которая включала ряд новых возможностей. После этого язык развивался относительно медленно вплоть до 2011 года. Но при этом в 1998 году была предпринята первая попытка по стандартизации языка организацией ISO (International Organiztion for Standartization). Первый стандарт получил название ISO/IEC 14882:1998 или сокращенно С++98. В дальнейшем в 2003 была издана новая версия стандарта C++03.

**Задание.**

**5.Процессы и потоки (Processes and Threads).**

Разработать программу, демонстрирующую механизмы подготовки, создания и управления процессами. Программа-оболочка должна уметь создавать процесс, ожидать завершения процесса в отдельном потоке, а также завершать запущенные дочерние процессы.

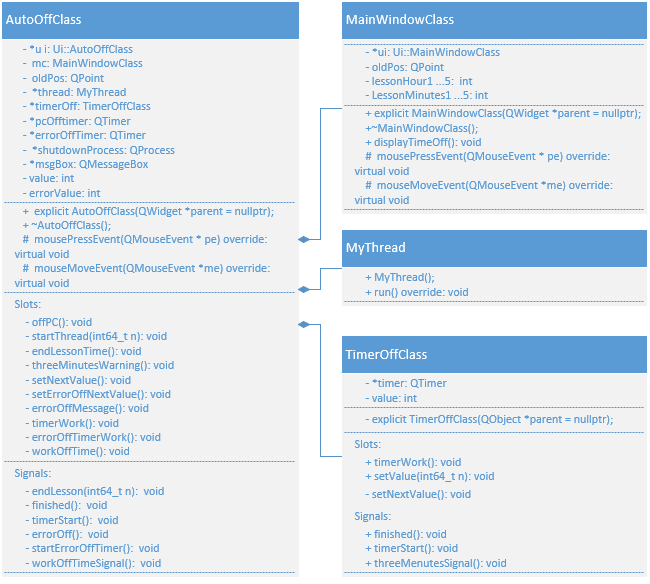




**Описание разработанных классов.**

Программа была создана с помощью фреймворка Qt в QtCreator. Всего реализовано 4 класса. Главный класс-агрегатор AutoOffClass (наследован от QWidget, в нем содержаться объекты остальных классов) и 3 вспомогательных класса MainWindowClass (также наследован от QWidget, класс основного окна программы), MyThread (наследован от QThread, отвечает за многопоточность), TimerOffClass (наследован от QObject, класс таймер работающий в параллельном потоке).

**Диаграмма классов.**

****

Описание программы.

Конструктор класса AutoOffClass.

Основная логическая нагрузка приходится на конструктор класса AutoOffClass (Рис.2) . Посредством конструктора задается стартовая заставка приложения (Рис.1):



Рис.1 С стартовая заставка приложения

Запускается вторичный поток, а так же прописаны функции connect для соединения сигналов со слотами:



Рис.2 Конструктор класса AutoOffClass.

Функция setAttribute (строки 9,10) задают прозрачность фона для основного виджета и элемента QLabel, в строке 11 по средством функции setPixmap мы задаем стартовое изображение которое находится в ресурсах программы. Строки с 13 по 19 это создание layout и кастомной кнопки завершения программы, сигнал которой связан функцией connect со слотом закрытия программы (строка 15).

В строке 21 инициализируется переменная типа QMessageBox для выведения на экран информации об ошибках и предупреждения. Далее инициализируется переменная типа QProcess (строка 22), которая отвечает за запуск процесса выключения компьютера в заданное время. Следом 2 переменных 2 таймера. В строках 27 и 28 инициализируются объекты наших классов thread (MyThread) и timerOff (TimerOffClass) и в строке 29 переменная timerOff передается в поток.

На строке 42 у нас находится функция endLesson (Рис.3). Она фиксирует текущее время при запуске компьютера ( при условии что компьютер запущен в рабочее время, иначе программа выводит предупреждение и закрывается) и посылает сигнал endLesson, по средством библиотечной функции connect (строка 32), слоту startThread (Рис.4).

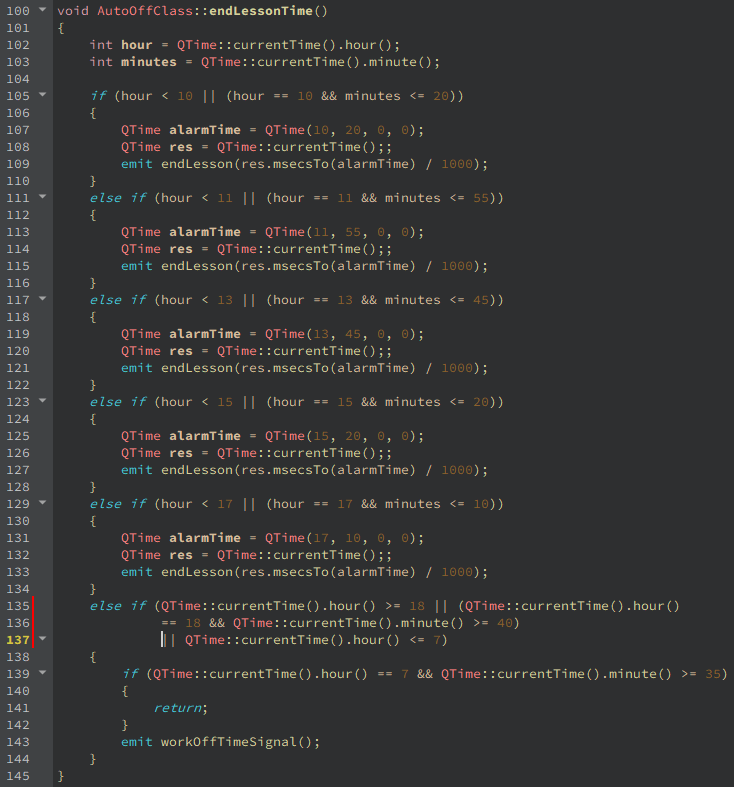


Рис.3 Функция endLesson.

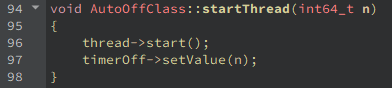


Рис.4 Слот startThread.

В данном случае логика следующая: из функции endLesson передается одноименный сигнал который передает слоту startThread количество секунд для ближайшей перемены (как получает эти данный сигнал можно увидеть например в строках 107-109 функции endLesson). Слот в свою очередь запускает параллельный поток с обратным таймером (класс TimerOffClass, файл cpp, Рис.5) который берет за время отсчета то самое количество секунд (передается через setValue строка 97, Рис.4).



Рис.5 Файл cpp класса TimerOffClass.

Далее работа программы продолжается в TimerOffClass. Функция setValue по средствам библиотечной функции connect (строка 8, Рис.5) посылает сигнал timerStart слоту timerWork, тем самым запускает таймер (строка 14, Рис.5). Таймер каждую секунду посылает сигнал timeout который через connect (строка 9, Рис.5) передается слоту setNextValue.

Если заглянуть в функцию setNextValue (строки 25-35, Рис.5) то мы увидим что переданное из класса AutoOffClass значение value каждую секунду уменьшается на единицу. И когда значение достигает нуля, это говорит нам что наступила перемена. Далее мы останавливаем таймер и посылаем 2 сигнала в класс AutoOffClass (строки 31,32, Рис.5).

Возвращаемся в конструктор класса AutoOffClass. Сигнал finished сообщает классу MyThred, по средством функции connect, что необходимо закрыть поток, так как он нам больше не нужен (строка 31, Рис.2). А сигнал threeMinutesSignal класса TimerOffClass через connect запускает слоты timerWork (строка 33, Рис.2) и threeMinutesWarning (строка 35, Рис.2) класса AutoOffClass. threeMinutesWarning показывает сообщение о том что через три минуты компьютер будет выключен (Рис.6).

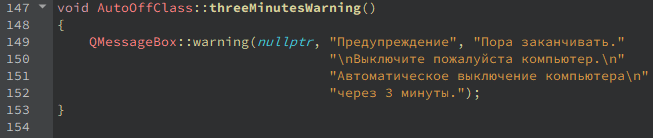


Рис.6 Слот threeMinutesWarning.

А слот timerWork запускает абсолютно аналогичный по логике реализации трехминутный таймер класса AutoOffClass который мы наблюдали в классе TimerOffClass. Сигнал таймаут каждую секунду вызывает слот setNextValue класса AutoOffClass (строка 34, Рис.2), в слоте значение value уменьшается на единицу и при значении ноль посылает сигнал finished который в свою очередь запускает (строка 36, Рис.2) слот offPC (Рис.7).

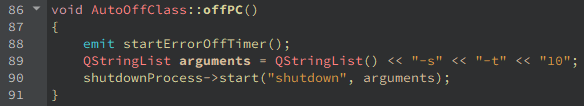


Рис.7 Слот offPC.

Слот offPC используя список аргументов (строка 89, Рис.7) запускает процесс выключения компьютера (строка 90, Рис.7).

Далее следует сказать об обработке ошибки выключения компьютера. Из слота offPC посылается сигнал startErrorOffTimer который запускает 20-секундный таймер (строка 38, Рис.2) с аналогичной логикой работы выше описанных таймеров, сигнал таймаут которого каждую секунду запускает слот setErrorOffNextValue (строка 39, Рис.2). При достижении нуля слот setErrorOffNextValue отправляет сигнал errorOff который запускает (строка 40, Рис.2) слот errorOffMessage (Рис.8). Этот слот сообщает о том что компьютер выключить не удалось.

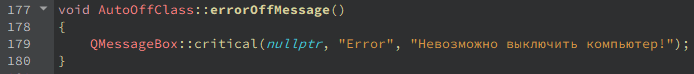


Рис.8 Слот errorOffMessage.

Обработка события включения компьютера в нерабочее время реализована в функции endLesson (строки 135-144). Отправляется сигнал workOffTimeSignal который запускает (строка 37, Рис.2) слот workOffTime (Рис.9)

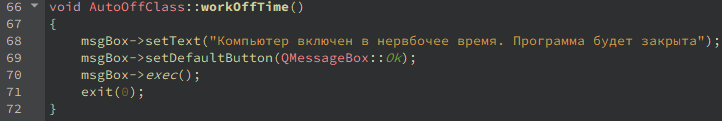


Рис.9 Слот workOffTime.

Слот выкидывает соответствующее сообщение и завершает программу.

Класс MainWindowClass отвечает лишь за выведение лишь актуальной информации когда выключится компьютер. За это отвечает функция displayTimeOff (Рис.11). Она предельно проста, поэтому в описании не нуждается. Так же здесь содержаться вспомогательные библиотечные функции mousePressEvent и mouseMoveEvent (Рис.10). Они отвечают за перемещение виджетов программы на рабочем столе. Абсолютно аналогичные функции содержаться и в классе AutoOffClass.

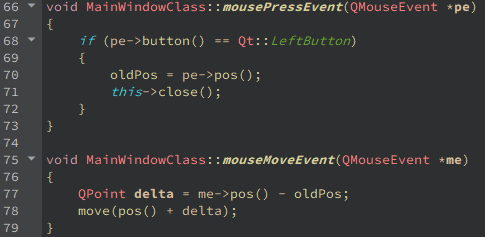


Рис.10 Функции mousePressEvent и mouseMoveEvent



Рис.11 Функция displayTimeOff.

Испытание программы.

При включении компьютера автозапуск запускает стартовое окно программы (Рис.1). При нажатии в любой точке стартового окна открывается основное окно программы (Рис.12).

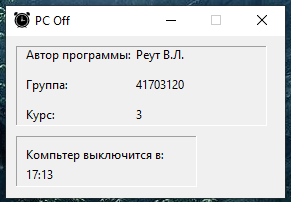


Рис.12 Основное окно программы.

Размер основного окна фиксирован согласно требованиям, а так же закрывается при щелчке левой кнопкой мышки в любом месте.

Согласно расписанию пар, поступает сообщении от том что компьютер будет выключен через 3 минуты (Рис.13).

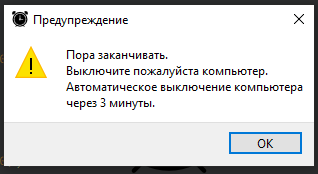


Рис.13 Предупреждение о выключении.

По истечении 3-х минут приходит системное предупреждение о выключении компьютера (Рис.14).

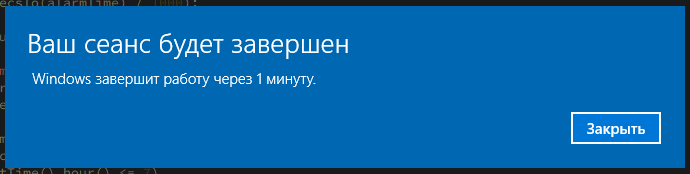


Рис.14 Системное предупреждение о выключении компьютера.

Если что-то мешает завершению работы то мы получаем сообщение о том что не удалось выключить компьютер (Рис.15)

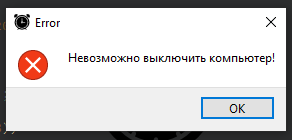


Рис.15 Сообщение о том что не удалось выключить компьютер.

Выводы.

В процессе создания приложения мы разобрались в работе потоков и процессов, а так же чем они отличаются. Так же оценили удобство работы с данными технологиями при использовании современных фреймворков на примере фреймворка Qt.

ЛИТЕРАТУРА

1. Qt 5.10 Профессиональное программирование на С++. Макс Шлее.
2. Документация Qt.

Листинг программы.