|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01 Информатика и вычислительная техника**

**Отчет(ВАРИАНТ 12)**

|  |  |
| --- | --- |
| **по домашней работе №** | **3** |

**Название:**

**Программирование на С++ с использованием классов**

**Дисциплина:** Объектно-ориентированное программирование

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-23Б |  | 14.05.2023 | В.К. Залыгин |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | А.М.Минитаева |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2023

**Цель работы**

Изучить основы работы с объектной моделью C++ и библиотеки графических интерфейсов QT.

**Задание**

Часть 3.1. Композиция

Разработать и реализовать диаграмму классов для описанных объектов предметной области, используя механизм композиции. Протестировать все методы каждого класса. Все поля классов должны быть скрытыми (private) или защищенными (protected). Методы не должны содержать операций ввода/вывода, за исключением процедуры, единственной задачей которой является вывод информации об объекте на экран.

Объект – продукт. Поля: название, дата изготовления, срок годности в днях. Методы: процедура инициализации, процедура вывода информации об объекте на экран, функции, возвращающие значения полей по запросу, и функция вычисления даты, до которой годен продукт.

**Проект программы**

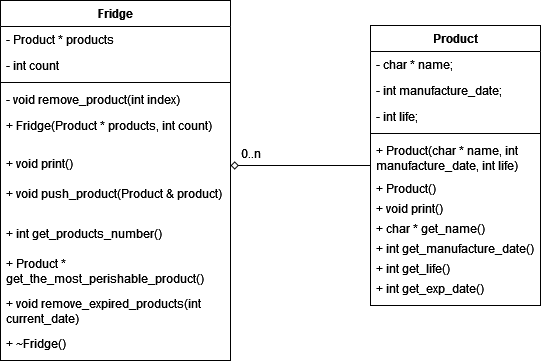
****

Рисунок 1 – диаграмма классов

**Текст программы**

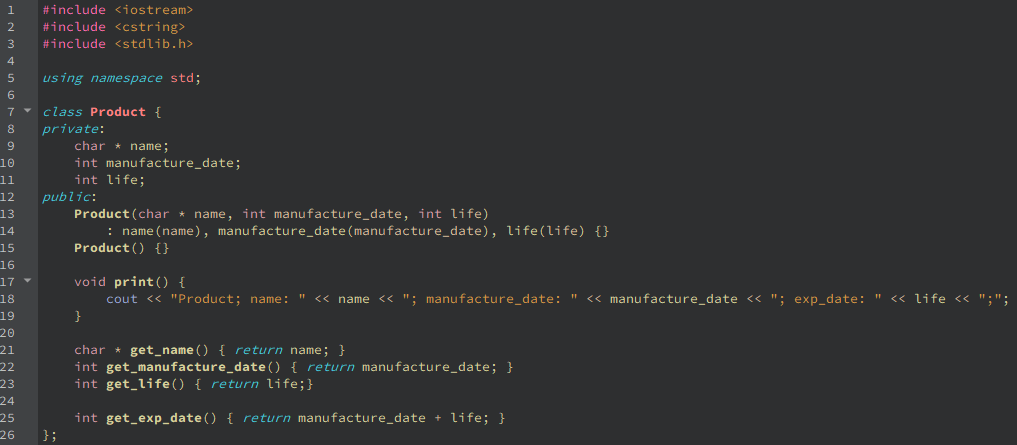


Рисунок 2 - код программы. Класс Product

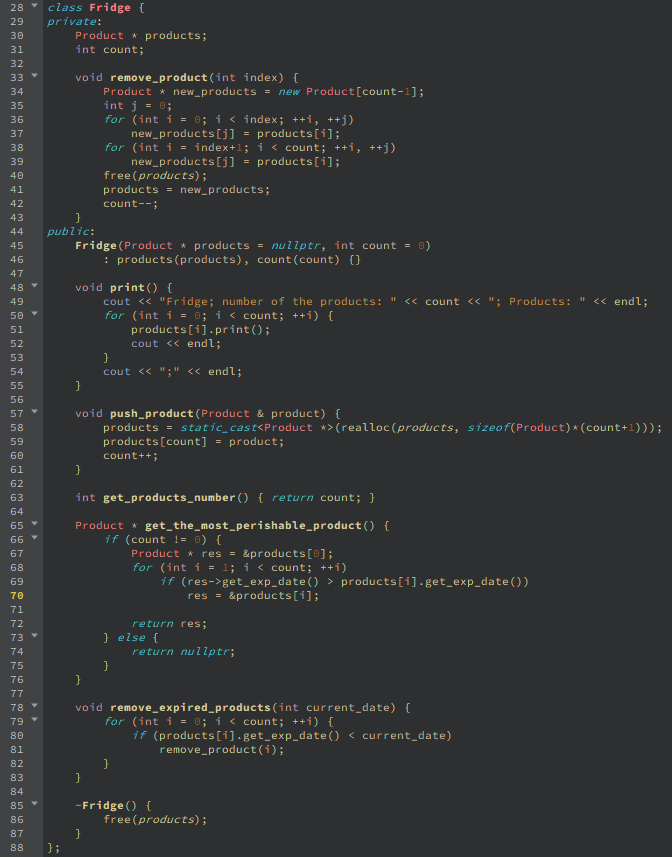


Рисунок 3 - код программы. Класс Fridge

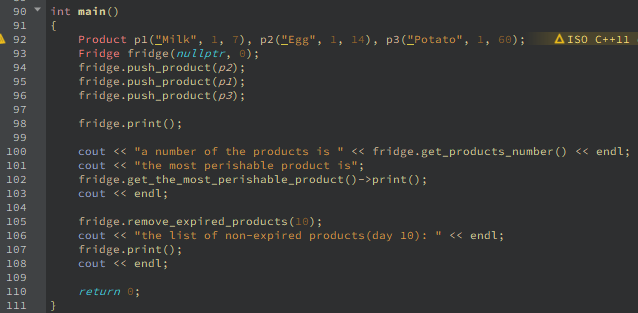


Рисунок 4 - код программы. Функция Main

**Тестовые данные**

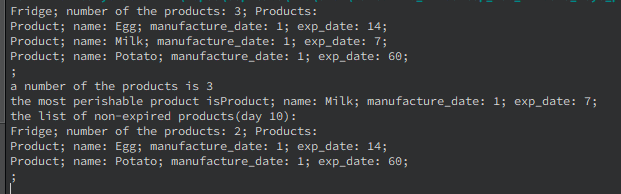


Рисунок 5 - результат работы

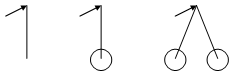
**Вывод**

Была разработана объектная декомпозиция задачи, реализована программа на языке C++ с использование базовых возможностей объектной модели.

**Часть 3.2. Qt. Полиморфное наследование**

**Задание**

Разработать программу, содержащую описание трех графических объектов:



Реализуя механизм полиморфизма, привести объекты в одновременное колебательное движение вокруг указанных точек с разными амплитудами и периодами колебаний.

В отчете привести диаграмму используемых классов Qt и разработанных классов, граф состояний пользовательского интерфейса и объектную декомпозицию.

**Проект программы**

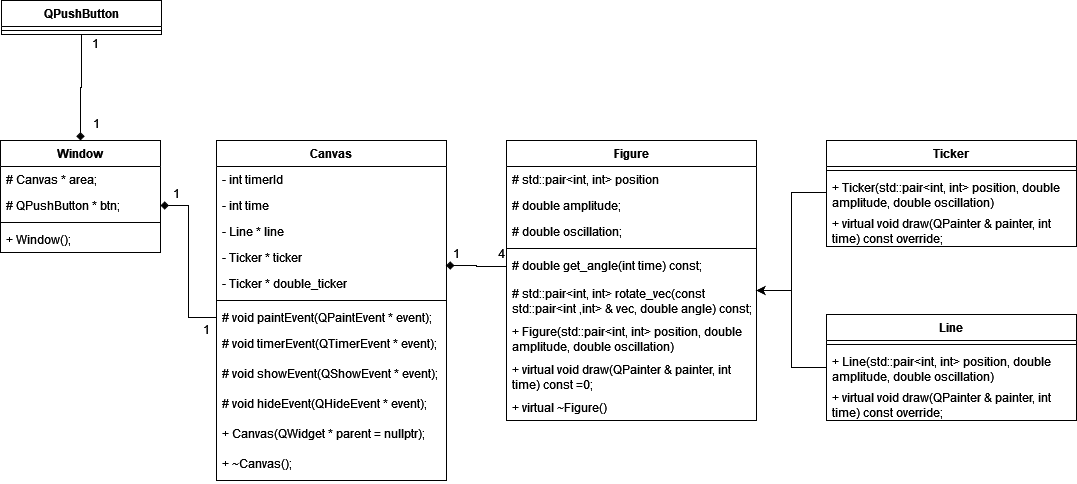
****

Рисунок 6 - диаграмма классов

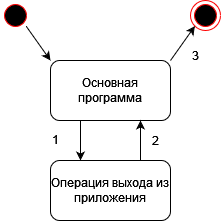


Рисунок 7 - граф состояний

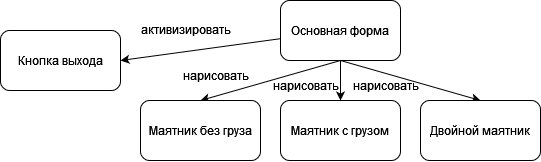


Рисунок 8 - объектная декомпозиция

**Текст программы**

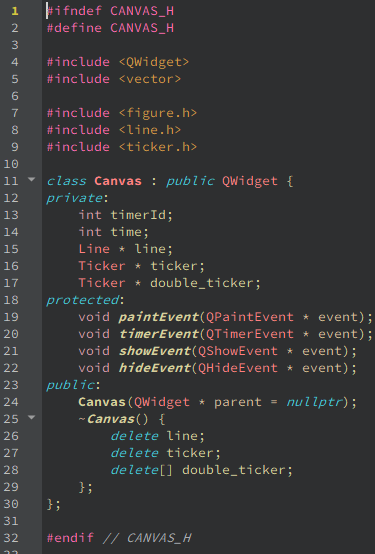


Рисунок 9 - код программы

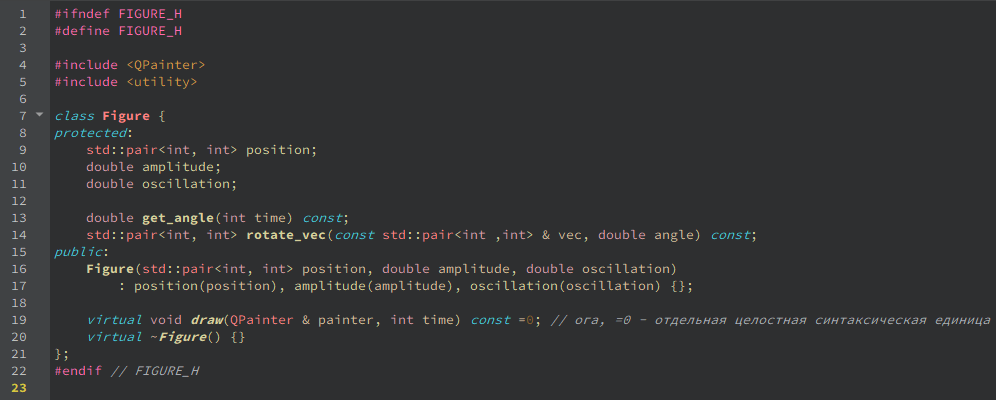


Рисунок 10 - код программы

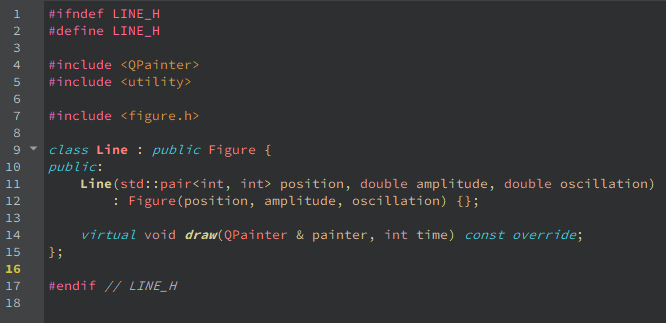


Рисунок 11 - код программы

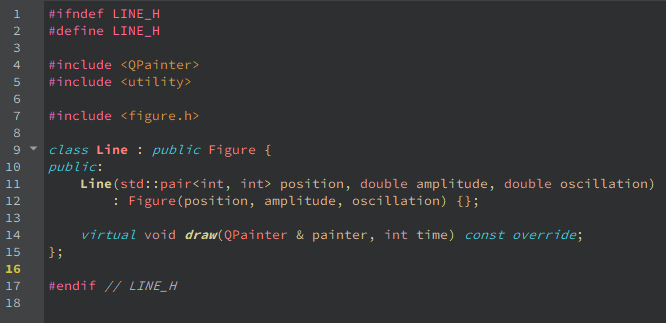


Рисунок 12 - код программы

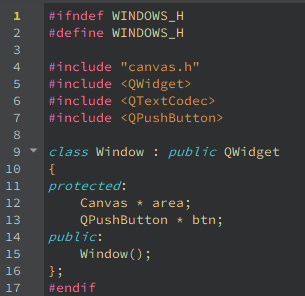


Рисунок 13 - код программы

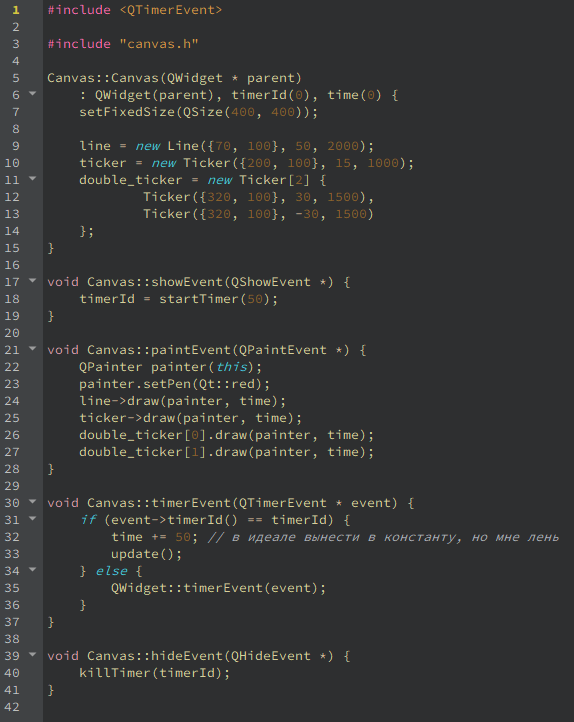


Рисунок - код программы

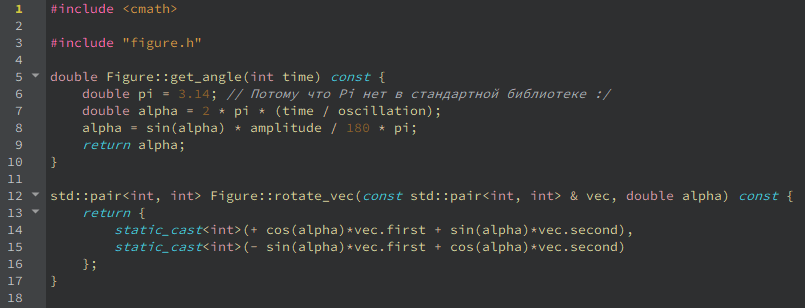


Рисунок 15 - код программы

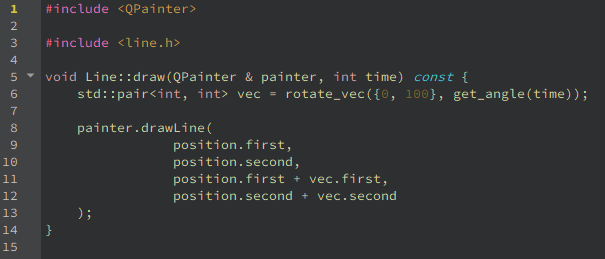


Рисунок 16 - код программы

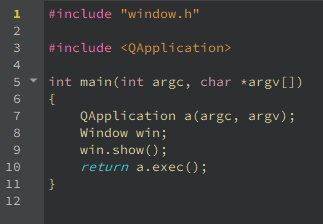


Рисунок 17 - код программы

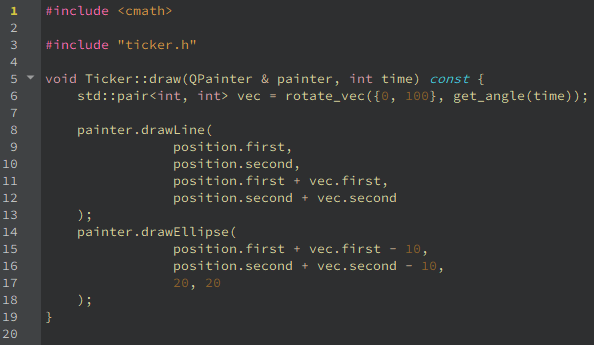


Рисунок 18 - код программы

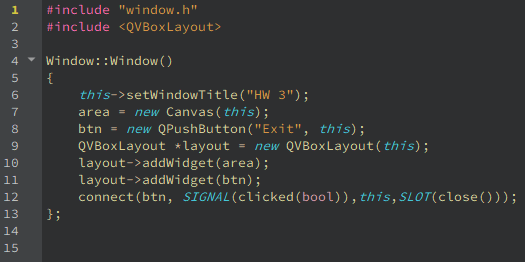


Рисунок 19 - код программы

**Тестовые данные**

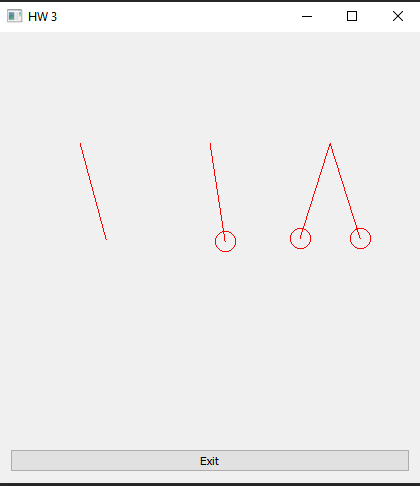


Рисунок 20 - пример работы

**Вывод**

Были изучены основы работы с QT.