МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное

учреждение высшего образования

«Ижевский государственный технический университет имени М. Т. Калашникова»

Факультет «Информатика и вычислительная техника»

Кафедра «Программное обеспечение»

Работа защищена с оценкой

«\_\_\_\_\_\_\_»

Дата\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КУРСОВАЯ РАБОТА

по дисциплине «Объектно-ориентированное программирование»

на тему «Система управления складом аптеки»

Выполнил:

Студент группы Б03-191-2з О.В.Кузнецов

Проверил:

Старший преподаватель К.С. Чернышев

Рецензия:

степень достижения поставленной цели работы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

полнота разработки темы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

уровень самостоятельности работы обучающегося\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

недостатки работы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Описание задания

Создать систему управления складом аптеки, которая реализует следующую функциональность:

1. Справочники: Группы лекарств, Лекарства, Производители, Поставщики
2. Функции: ведение справочников, учет прихода и продаж лекарств.
3. Создание документов: Отчет по наличию лекарств на складе по группам, отчет по продажам по группам, счет-фактура

Анализ предметной области

Предметная область поставленной задачи состоит из следующих объектов.

1. Поставщики и производители, которых можно объединить в общий термин фирмы. Для описания объекта фирмы необходимо указать название фирмы и ее адрес. Данный объект реализует методы, которые дают доступ к чтению названия фирмы и ее адрес.
2. Лекарство. Для его описания необходимы следующие данные: название лекарства, производителя, поставщика и также его цена. Объект лекарство содержит методы, дающие доступ к чтению его данных.
3. Группа лекарств. Состоит из названия группы и двух словарей, которые хранят сведения о поступлении и продаже лекарств данной группы. В первом словаре хранится словарь, в котором указаны количество имеющихся лекарств на складе, во втором количество проданных лекарств. Объект группа не реализует никакой функциональности и служит лишь для объединения двух словарей.
4. Аптека. Данный объект состоит из словарей, которые хранят поставщиков и производителей, от которых разрешено принимать поставки лекарств, ключом является название фирмы, а значением объект фирма. Также словарь из лекарств, которые реализуются в данной аптеке, значением данного словаря является объект лекарство, а ключом строка, состоящая из названия лекарства, производителя и поставщика. Такой вид ключа необходим, так как возможен случай, что одно и то же лекарство может выпускаться разными производителями или поставляться разными поставщиками по различным ценам. И словарь из групп, который разделяет имеющиеся лекарства на категории. Объект аптека реализует основную функциональность, необходимую для решения задачи: добавление нового производителя, поставщика, лекарства, группы, учет продажи и прихода лекарств, печать отчетов о продаже лекарств, об остатке на складе, счета-фактуры.

В словарь предметной области входят такие существительные как: фирма, поставщик, производитель, лекарство, группа, аптека. С аптекой связаны следующие глаголы: добавление нового элемента, печать отчета, продажа, поступление.

Диаграмма классов

|  |
| --- |
| **Pharmacy** |
| - producers : map<string, Firm>  - providers : map<string, Firm>  - meds : map<string, Med>  - groups : map<string, Group> |
| + downloadData ( )  + addGroup (group\_name : string)  + addProvider (provider\_name : string, address : string)  + addProducer (producer\_name : string, address : string)  + addMed (name : string, price : int, group\_name : string, producer\_name : string,  provider\_name : string)  + arrivalMedicament (med\_name : string, count : int, provider\_name : int,  producer\_name : string)  + printReserveReport ( )  + saleMedicament (med\_name : string, sale\_count : int, provider\_name : string,  producer\_name : string)  + printSaleReport ()  + printInvoiceReport ( provider\_name : string) |

|  |
| --- |
| **Firm** |
| - name : string  - address : string |
| + getName ( )  + getAddress ( ) |

|  |
| --- |
| **Group** |
| + name : string  + med\_count : map<string, int>  + med\_sold : map<string, int> |
|  |

|  |
| --- |
| **Med** |
| - name : string  - producer\_name : string  - provider\_name : string  - group\_name : string  - price : int |
| + getName ( )  + getProducerName ( )  + getProviderName ( )  + getGroupName ( )  \_ getPrice ( ) |

Выбор и обоснование языка

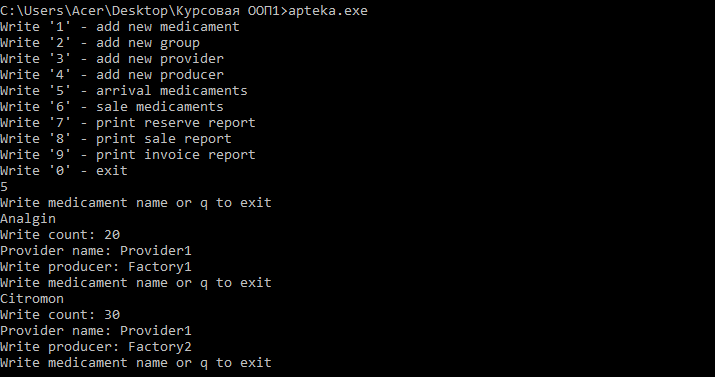
Для выполнения курсовой работы был выбран язык программирования С++. Из главных преимуществ этого языка можно выделить следующие.

Язык С++ поддерживает объектно-ориентированной подход, благодаря чему, создаваемую систему управления можно разделить на отдельные модули и при этом программа остается поддерживаемой и читаемой, что позволяет при необходимости вносить изменения в программный код с меньшей сложностью, чем при использовании других подходов программирования, таких как структурное программирование.

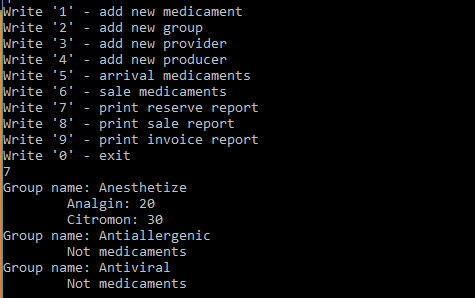
В языке С++ есть стандартная библиотека шаблонов, которая содержит самые часто используемые обобщенные алгоритмы и структуры данных. С ее помощью повышается производительность работы программиста, а код становится более надежным, коротким и читаемым. В фундаменте библиотеки лежат алгоритмы и структуры данных. Механизм работы STL очень прост и заключается в том, что алгоритмы не используют информацию о внутреннем устройстве структур данных, а обращаются к ним с помощью абстрактных методов итераторов, реализация которых регулируется набором строгих правил.

Контрольный пример

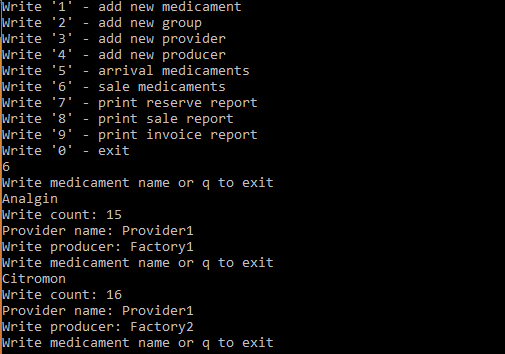
Поступление 20 единиц «Analgin» от поставщика «Provider1» и производителя «Factory1» и 30 единиц «Citromon» от поставщика «Provider1» и производителя «Factory2»



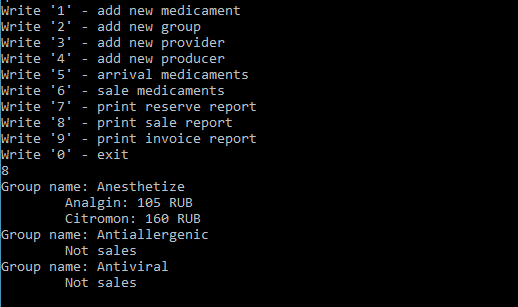
Печать отчета об остатках на складе по группам лекарств.



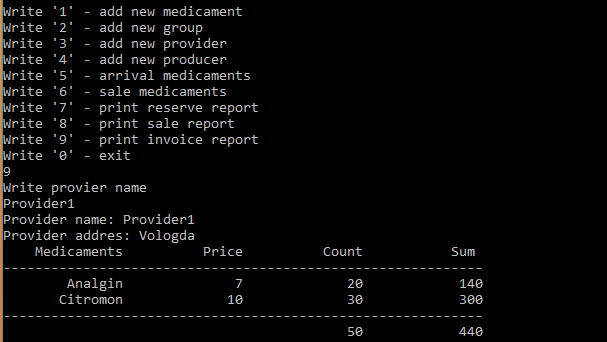
Продажа 15 единиц «Analgin» от поставщика «Provider1» и производителя «Factory1» и 16 единиц «Citromon» от поставщика «Provider1» и производителя «Factory2»



Печать отчета по продажам по группам.



Печать счета фактуры для поставщика «Provider1»



Код программы

class Firm {

private:

std::string name;

std::string address;

public:

Firm() {}

Firm(const std::string name, const std::string address);

std::string getName() { return name; }

std::string getAddress() { return address; }

};

class Med {

private:

std::string name;

std::string producer\_name;

std::string provider\_name;

std::string group\_name;

int price;

public:

Med() {}

Med(const std::string name,

int price,const std::string group\_name,

const std::string producer\_name,

const std::string provider\_name

);

std::string getName() { return name; }

std::string getProducerName() { return producer\_name; }

std::string getProviderName() { return provider\_name; }

std::string getGroupName() { return group\_name; }

int getPrice() { return price; };

};

class Group {

public:

std::string name;

std::map<std::string, int> med\_count;

std::map<std::string, int> med\_sold;

public:

Group() {}

Group(const std::string name) { this->name = name; }

};

class Pharmacy {

private:

std::map<std::string, Firm> producers;

std::map<std::string, Firm> providers;

std::map<std::string, Med> meds;

std::map<std::string, Group> groups;

public:

Pharmacy() { downloadData(); }

void downloadData();

void addGroup(const std::string group\_name);

void addProvider(const std::string provider\_name, const std::string address);

void addProducer(const std::string producer\_name, const std::string address);

bool addMed(

const std::string name,

int price,

const std::string group\_name,

const std::string producer\_name,

const std::string provider\_name

);

bool arrivalMedicament(

std::string med\_name,

int count,

std::string provider\_name,

std::string producer\_name

);

void printReserveReport();

bool saleMedicament(

std::string med\_name,

int sale\_count,

std::string provider\_name,

std::string producer\_name

);

void printSaleReport();

bool printInvoiceReport(std::string provider\_name);

};