|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**КУРСОВОЙ ПРОЕКТ (РАБОТА)**

по дисциплине:Объектно-ориентиронное програмирование

по профилю:Информация организация

направления профессиональной подготовки:прикладная информатика бакалавриат

Тема:«Программа для управления деятельностью склада торгового предприятия»

Студент: Скотников Владимир Вячеславович

Группа: ИНБО-04-18

Работа представлена к защите\_\_\_\_\_\_\_\_\_\_\_(дата)\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

(подпись и ф.и.о. студента)

Руководитель: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(Ассистент Кафедры ИиППО Хлебникова В.Л.)

Работа допущена к защите\_\_\_\_\_\_\_\_\_\_\_(дата)\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

(подпись ф.и.о. руководителя)

Оценка по итогам защиты: \_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /

(подписи, дата, ф.и.о., должность, звание, уч. степень двух преподавателей, принявших защиту)

М. МИРЭА. 2019г.

УДК004.432.4

Скотников В.В“Программа для управления деятельностью склада торгового предприятия” / Очёт по «объектно-ориентированное программированию» профиля **«**Информатизация Организаций**»** направления профессиональной подготовки бакалавриата 09.03.03 «Прикладная информатика» (2 ой семестр) / руководитель асс. Хлебникова В.Л./ кафедра ИППО Института ИТ МИРЭА – с. 19, илл. 1, ист. 3, (в т.ч. 1 на англ. яз.).

Цель работы “Программа для управления деятельностью склада торгового предприятия” представляет собой специальную программу, которую вы сможете использована для поддержания записей о товарах, имеющихся на складе в актуальном состоянии.

Каждая запись содержит сведения о конкретном товаре, например наименование, количество, срок годности и т.д.

The purpose of the work “to Develop a program for managing the activities of the warehouse of a trading enterprise” is a special program that you can use to maintain records of goods available in the warehouse in the current state.

Each record contains information about a specific item, such as name, quantity, expiration date, etc.

М. МИРЭА. Ин-т ИТ. Каф. ИиППО. 2019г. @ В.Скотников

|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**ЗАДАНИЕ НА ВЫПОЛНЕНИЕ КУРСОВОГО ПРОЕКТА (РАБОТЫ)**

по дисциплине: объектно-ориентированное программированию

по профилю: Информатизация Организаций

направления профессиональной подготовки:Прикладная информатика (академический/прикладной бакалавират)

Студент (ф.и.о. полностью):Скотников Владимир Вячеславович

Группа:ИНБО-04-18

Срок представления к защите \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель: ассистент кафедры Хлебникова В.Л.

Тема: «Программа для управления деятельностью склада торгового предприятия»

Исходные данные: документы и средства языка C++

Перечень вопросов к разработке (основная задача), графические материалы (слайды):Добавление нового товара в колекцию записей,Редактирование имеющейся информации о товаре,Удаление записей о товаре,Обеспечевать возможность сортировки товаров в алфовитном порядке по названию

Заведующий кафедрой ИиППО:\_\_\_\_\_\_\_\_\_\_ /В.А. Мордвинов/, дата:\_\_\_\_\_\_\_\_

Задание на КП (КР) выдал\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (должн. и Ф.И.О. руководителя)

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_201\_\_г.

Задание на КП (КР) получил\_\_\_\_\_\_\_\_\_\_\_\_\_ (Ф.И.О. студента)

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_201\_\_г.

Оглавление

[1.1.Введение 4](#_Toc9441563)

[1.1.1.Наименование программы 4](#_Toc9441564)

[1.1.2.Назначение и область применения 4](#_Toc9441565)

[1.2.Требования к программе или программному изделию 4](#_Toc9441566)

[1.2.1.Требования к функциональным характеристикам 4](#_Toc9441567)

[1.3.Условия эксплуатации 4](#_Toc9441568)

[1.3.1.Требования к составу и параметрам технических средств 4](#_Toc9441569)

[1.3.2.Требования к информационной и программной совместимости 5](#_Toc9441570)

[1.4.Требования к программной документации 5](#_Toc9441571)

[1.4.1.Предварительный состав программной документации 5](#_Toc9441572)

[1.5.Стадии и этапы разработки 5](#_Toc9441573)

[1.5.1.Стадии разработки 5](#_Toc9441574)

[1.5.2.Этапы разработки 5](#_Toc9441575)

[1.5.3.Содержание работ по этапам 6](#_Toc9441576)

[1.6.Порядок контроля и приемки 7](#_Toc9441577)

[1.6.1.Виды испытаний 7](#_Toc9441578)

[1.6.2.Общие требования к приемке работы 7](#_Toc9441579)

[Глава 2. Проектирование и разработка программы 8](#_Toc9441580)

[2.1.Проектирование программы 8](#_Toc9441581)

[2.1.1Функциональные требования к программе 8](#_Toc9441582)

[2.1.2.Проектирование классов 8](#_Toc9441583)

[2.1.3.Описание проектного решения 8](#_Toc9441585)

[2.1.1.Тестирование программы 9](#_Toc9441586)

[Глава 3. Руководство пользователя 10](#_Toc9441587)

[Заключение 11](#_Toc9441588)

[Список литературы 12](#_Toc9441591)

[Приложение №1. Исходный код программы 13](#_Toc9441592)

## ВВЕДЕНИЕ

В ходе данного курсовой работы должны будут закрепиться знания по Объективно-Ориентированному Программированию на языке программирования С++. C++ —В частности: поддерживаются различные стили и технологии программирования, включая традиционное директивное программирование, ООП, обобщённое программирование.

“Программа для управления деятельностью склада торгового предприятия” представляет собой специальную программу, которую вы сможете использована для поддержания записей о товарах, имеющихся на складе в актуальном состоянии.

Каждая запись содержит сведения о конкретном товаре, например наименование, количество, срок годности и т.д.

### Наименование программы

Наименование программы – «Программа для управления деятельностью склада торгового предприятия».

### Назначение и область применения

Программа «Программа для управления деятельностью склада торгового предприятия» предназначена для поддержания записей о товарах, имеющихся на складе в актуальном состоянии.

## Требования к программе или программному изделию

### Требования к функциональным характеристикам

Программа должна обеспечивать возможность выполнения перечисленных ниже функций:

1. Добавление нового товара в колекцию записей.
2. Редактирование имеющейся информации о товаре.
3. Удаление записей о товаре.
4. Обеспечевать возможность сортировки товаров в алфовитном порядке по названию.

## Условия эксплуатации

### Требования к составу и параметрам технических средств

В состав технических средств должен входить IВМ-совместимый персональный компьютер (ПЭВМ), включающий в себя:

1. Процессор Pentium-2.0Hz, не менее;   
2. Оперативную память объемом, 1 Гигабайт, не менее;   
3. Любую операционную систему;   
4. Стандартный пакет С++;

### Требования к информационной и программной совместимости

#### Требования к исходным кодам и языкам программирования

Исходные коды программы должны быть реализованы на языке C++.

#### Требования к программным средствам, используемым программой

Системные программные средства, используемые программой, должны быть представлены лицензионной локализованной версией любой операционной системы и ПО C++.

## Требования к программной документации

### Предварительный состав программной документации

Состав программной документации должен включать в себя:

1. техническое задание;
2. программу и методики испытаний;
3. руководство пользователя.

## Стадии и этапы разработки

### Стадии разработки

Разработка должна быть проведена в три стадии:

1. разработка технического задания;
2. рабочее проектирование;
3. внедрение.

### Этапы разработки

На стадии разработки технического задания должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

1. разработка программы;
2. разработка программной документации;
3. испытания программы.

На стадии внедрения должен быть выполнен этап разработки - подготовка и передача программы.

### Содержание работ по этапам

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:

1. постановка задачи;
2. определение и уточнение требований к техническим средствам;
3. определение требований к программе;
4. определение стадий, этапов и сроков разработки программы и документации на неё;
5. согласование и утверждение технического задания.

На этапе разработки программы должна быть выполнена работа по программированию (кодированию) и отладке программы.

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями к составу документации.

На этапе испытаний программы должны быть выполнены перечисленные ниже виды работ:

1. разработка, согласование и утверждение программы и методики испытаний;
2. проведение приемо-сдаточных испытаний;
3. корректировка программы и программной документации по результатам испытаний.

На этапе подготовки и передачи программы должна быть выполнена работа по подготовке и передаче программы и программной документации в эксплуатацию.

## Порядок контроля и приемки

### Виды испытаний

Демонстрация работы программы и ее функционала.

### Общие требования к приемке работы

Программа должна соответствовать всем вышесказанным требованиям.

# ГЛАВА 2. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПРОГРАММЫ

## Проектирование программы

### Функциональные требования к программе

### Функциональные требования к программе описаны в пункте № 1.2 Технического задания.

## Проектирование классов

Для реализации проекта будет использован язык программирования C++. Данный язык является одним из самых наиболее распространённых объектно-ориентированных языков программирования (ООП).

# Целью данной работы является написание программы “ Программа для управления деятельностью склада торгового предприятия ”, используя объектно-ориентированное программирование. Написание программы способствовало закреплению теоретического материала на практических занятиях.

Разработка программы

## Описание проектного решения

На рисунке1.1 придставлена общая схема диаграмма

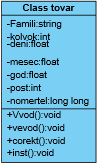
 

Рисунок 1.1 – Общая UML диаграмма

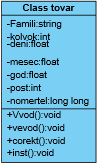


Рисунок.1.1. UML диаграмма класса Tovar

Tovar –класс отвечающий за ввод, вывод,корекцию данных.



Рисунок.1.2. UML диаграмма класса ddaattaa

Ddaattaa-класс овечающий за ввод и вывод текущей даты.

## Тестирование программы

Тестирование программы производилось путем проверки выполнения корректности работы программы. В нормальных условиях программа работала без каких-либо сбоев. Все необходимые операции выполняются верно.

# ГЛАВА 3. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Запуск программного приложения осуществляется через исполняемый файл Скотников склад ТТ.exe.

Программное приложение следует запускать двойным нажатием правой кнопки мыши по исполняемому файлу Скотников склад TT.exe.

После запуска программного приложения будет открыто главное меню, в котором следует ввести текущую дату и начать ввод информации о товарах.

Программное приложения позволяет производить добавление, сохранение, удаление и изменение инфорации о товарах.

Просроченые товары автоматически не учитываются в конечном списке товаров.

# ЗАКЛЮЧЕНИЕ

# В результате выполнения курсовой работы была реализована программное приложение для управления деятельностью склада торгового предприятия, с используованием объектно-ориентированного языка программирования С++.

# Разработка программного приложения производилась с использованием учебных пособий, указанных в списке информационных источников, указанных в

# СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ

1. Т. А. Павловская. C/C++. Программирование на языке высокого уровня. Из серии "300 лучших учебников".2003 год. 461 стр.
2. Прата Стивен. Язык программирования С++. Лекции и упражнения. Учебник. 2005 год. 1100 стр.
3. Шлее Макс - Профессиональное программирование на C++. +CD. Qt 4.8. 2012

# Приложение №1. Исходный код программы

#include "pch.h"

#include <iostream>

#include <string>

#include <fstream>

#include <vector>

#include <cmath>

using namespace std;

class tovar

{

public:

string Famili;

int kolvok;

float deni;

float mesec;

float god;

long long nomertel;

int pros;

int nomer;

void Vvod()

{

cout << "название товара" << endl;

cin >> Famili;

cout << "количество" << endl;

cin >> kolvok;

cout << "введите годен до(день,месяц,год)" << endl;

cout << "введите день" << endl;

cin >> deni;

cout << "введите месяц" << endl;

cin >> mesec;

cout << "введите год" << endl;

cin >> god;

cout << "введите номер (телефона)" << endl;

cin >> nomertel;

pros = 0;

}

void vevod()

{

cout << Famili << " "

<< kolvok << " "

<< deni << "." << mesec << "." << god << " " <<

nomertel << endl;

}

void nol()

{

Famili = "a";

kolvok = 0;

deni = 0;

mesec = 0;

god = 0;

nomertel = 0;

//pros = 1;

}

void corekt()

{

int p;

cin >> p;

if ((p > 0) && (p < 2))

{

cout << "новое название товара: " << endl;

cin >> Famili;

}

if ((p > 1) && (p < 3))

{

cout << "новое количество" << endl;

cin >> kolvok;

}

if ((p > 2) && (p < 4))

{

cout << "введите годен до: " << endl;

cout << "введите день" << endl;

cin >> deni;

cout << "введите месяц" << endl;

cin >> mesec;

cout << "введите год" << endl;

cin >> god;

}

if ((p > 3) && (p < 5))

{

cout << "введите новый номер телефона: " << endl;

cin >> nomertel;

}

}

void inst()

{

cout << "инструкция z" << endl;

char er;

cin >> er;

if (er == 'z')

{

system("start инс.txt");

}

}

string& getFamili()

{

return Famili;

}

int& getKolvok()

{

return kolvok;

}

float& getDeni()

{

return deni;

}

float& getMesec()

{

return mesec;

}

float& getGod()

{

return god;

}

long long& getNomertel()

{

return nomertel;

}

void setFamili(string Famili)

{

this->Famili = Famili;

}

void setKolvok(int kolvok)

{

this->kolvok = kolvok;

}

void setDeni(float deni)

{

this->deni = deni;

}

void setMesec(float mesec)

{

this->mesec = mesec;

}

void setGod(float god)

{

this->god = god;

}

void setNomertel(long long nomertel)

{

this->nomertel = nomertel;

}

};

class ddaattaa

{

public:

float deni;

float mesec;

float god;

void vvod()

{

cout << "введите день" << endl;

cin >> deni;

cout << "введите месяц" << endl;

cin >> mesec;

cout << "введите год" << endl;

cin >> god;

}

void vevod()

{

cout << deni << "."

<< mesec << "."

<< god << " "

<< endl;

}

};

int main()

{

string data;

int i, conekt, c, l, p, io, io1;

p = 0;

int h, h1, r, port, sohr, sohr1;

port = 0;

h = 0;

c = 0;

sohr = 0;

sohr1 = 0;

conekt = 0;

r = 0;

io1 = 0;

l = 0;

float syma, syma1;

int n, a, hoor, x, y, z, o;

int k;

k = 1;

hoor = 0;

setlocale(LC\_ALL, "Russian");

tovar carta[400];

ddaattaa now;

for (int i = 0; i < 400; i++)

{

carta[i].kolvok = 0;

}

carta[400].inst();

ifstream lol1("test1.txt");

lol1 >> io1;

lol1.close();

cout << "введите дату(день,месяц,год) " << endl;

now.vvod();

syma = (now.deni / 100000) + (now.mesec / 100) + now.god;

while (hoor < 2)

{

if (io1 != 0)

{

ifstream lol("test1.txt");

lol >> sohr1;

lol.close();

}

io1 = io1 + 1;

carta[400].inst();

if (sohr1 == 0)

{

cout << "введите кол-во карт" << endl;

cin >> n;

for (int i = 0; i < n; i++)

{

syma1 = 0;

carta[i].Vvod();

syma1 = (carta[i].deni / 100000) + (carta[i].mesec / 100) + carta[i].god;

if (syma >= syma1)

{

carta[i].pros = 1;

}

}

a = 0;

x = 0;

y = 0;

int l = 1;

while (l != 0)

{

l = 0;

for (int i = 0; i < n - 1; i++)

{

if (carta[i].Famili > carta[i + 1].Famili)

{

l = l + 1;

swap(carta[i].Famili, carta[i + 1].Famili);

swap(carta[i].kolvok, carta[i + 1].kolvok);

swap(carta[i].deni, carta[i + 1].deni);

swap(carta[i].mesec, carta[i + 1].mesec);

swap(carta[i].god, carta[i + 1].god);

swap(carta[i].nomertel, carta[i + 1].nomertel);

swap(carta[i].pros, carta[i + 1].pros);

}

}

}

}

else

{

ifstream file("test.txt");

vector<tovar> array;

tovar test;

for (int i = 0; i < sohr1; i++)

{

string Famili, kolvok, deni, mesec, god, nomertel;

getline(file, Famili, ' ');

getline(file, kolvok, ' ');

getline(file, deni, ' ');

getline(file, mesec, ' ');

getline(file, god, ' ');

getline(file, nomertel);

test.setFamili(Famili);

test.setKolvok(stoi(kolvok)); // флоат stof

test.setDeni(stof(deni));

test.setMesec(stof(mesec));

test.setGod(stof(god));

test.setNomertel(stoll(nomertel)); // инт stoi

array.push\_back(test);

}

file.close();

for (int i = 0; i < sohr1; i++)

{

cout << i << ": " << endl << "Famili: " << array[i].getFamili() << endl <<

"Kolvok: " << array[i].getKolvok() << endl <<

"Deni: " << array[i].getDeni() << "Mesec: " << array[i].getMesec() << "God: "

<< array[i].getGod() << "nomertel: " << array[i].getNomertel() << endl;

carta[i].Famili = array[i].getFamili();

carta[i].kolvok = array[i].getKolvok();

carta[i].deni = array[i].getDeni();

carta[i].mesec = array[i].getMesec();

carta[i].god = array[i].getGod();

carta[i].nomertel = array[i].getNomertel();

carta[i].pros = 0;

}

}

if (sohr1 == 0)

{

for (int i = 0; i < n; i++)

{

cout << i << ")";

carta[i].vevod();

}

}

else

{

for (int i = 0; i < sohr1; i++)

{

cout << i << ")";

carta[i].vevod();

}

}

cout << endl;

cout << "введите 1 чтобы завершить работу " << "перерасчет 0" << endl;

cin >> x;

if (x == 0)

{

cout << "введите 1 изменить иформацию(удалить) о товаре(существующем) " << "добавить в базу данных 0" << endl;

cin >> y;

if (y == 0)

{

r = 0;

h = 0;

while (r < 1)

{

cout << "Сколько элементов нужно добавить " << endl;

cin >> a;

h = h + a;

if (sohr1 == 0)

{

c = n + a;

for (int i = n; i < c; i++)

{

syma1 = 0;

carta[i].Vvod();

syma1 = (carta[i].deni / 100000) + (carta[i].mesec / 100) + carta[i].god;

if (syma >= syma1)

{

carta[i].pros = 1;

}

}

swap(n, c);

}

else

{

c = sohr1 + a;

for (int i = sohr1; i < c; i++)

{

syma1 = 0;

carta[i].Vvod();

syma1 = (carta[i].deni / 100000) + (carta[i].mesec / 100) + carta[i].god;

if (syma >= syma1)

{

carta[i].pros = 1;

}

}

swap(sohr1, c);

}

cout << "1)завершить 0)продолжить " << endl;

cin >> r;

}

}

else

{

z = 0;

while (z < 2)

{

cout << "хотите удалить товар введите 1 изменить инф введите 0" << endl;

cin >> port;

if (port < 1)

{

cout << "введите номер товара " << endl;

cin >> i;

cout << "введите номер элемента который нужно изменить" << endl;

cout << "1)название товара: " << carta[i].Famili << " "

<< "2)количество: " << carta[i].kolvok << " "

<< "3)введите годен до: " << carta[i].deni << " " <<

carta[i].mesec << " " << carta[i].god << " "

<< "4)введите телефона: " << carta[i].nomertel << endl;

carta[i].corekt();

}

else

{

int aa;

aa = 0;

cout << "введите номер товара " << endl;

cin >> aa;

carta[aa].kolvok = 0;

}

cout << "введите 1)продолжить 2)завершить корекцию" << endl;

cin >> z;

}

}

if (sohr1 == 0)

{

c = n + h;

}

else

{

c = sohr1 + h;

}

l = 1;

while (l != 0)

{

l = 0;

for (int i = 0; i < c - 1; i++)

{

if (carta[i].Famili > carta[i + 1].Famili)

{

l = l + 1;

swap(carta[i].Famili, carta[i + 1].Famili);

swap(carta[i].kolvok, carta[i + 1].kolvok);

swap(carta[i].deni, carta[i + 1].deni);

swap(carta[i].mesec, carta[i + 1].mesec);

swap(carta[i].god, carta[i + 1].god);

swap(carta[i].nomertel, carta[i + 1].nomertel);

swap(carta[i].pros, carta[i + 1].pros);

}

}

}

}

if (sohr1 == 0)

{

c = n + h;

}

else

{

c = sohr1 + h;

}

cout << endl;

ofstream rewrite("test.txt");

rewrite.write("", 0);

rewrite.close();

ofstream rewrite1("test1.txt");

rewrite1.write("", 0);

rewrite1.close();

fstream f;

f.open("test.txt");

sohr = 0;

for (int i = 0; i < c; i++)

{

if ((carta[i].kolvok != 0) && (carta[i].kolvok >= 1) && (carta[i].pros == 0))

{

cout << i << ")";

carta[i].vevod();

f << carta[i].Famili << " "

<< carta[i].kolvok << " "

<< carta[i].deni << " " << carta[i].mesec << " " << carta[i].god << " " <<

carta[i].nomertel << endl;

sohr = sohr + 1;

}

}

f.close();

fstream fe;

fe.open("test1.txt");

fe << sohr;

fe.close();

cout << endl;

cout << "введите 1)продолжить корекцию 2)завершить работу" << endl;

cin >> hoor;

for (int i = 0; i < 400; i++)

{

carta[i].nol();

}

}

system("pause");

return 0;

}