Министерство образования Республики Беларусь

Учреждение образования

“Брестский государственный университет”

Кафедра ИИТ

Лабораторная работа №1

По дисциплине: “ЯП”

Тема: “Классы и объекты в С++”

Выполнил:

Студент 2-го курса

Группы ПО-7

Смушко О.Р.

Дата выполнения: 15.09.21

Проверил:

Бойко Д.О.

Брест, 2021

**Вариант 8**

Задание.

Определить пользовательский класс ТОВАР с полями: имя (string), количество (int), стоимость (float). Определить в классе следующие конструкторы: с параметрами, без параметров, копирования. Определить в классе деструктор. Определить в классе компоненты-функции для просмотра и установки полей данных. Определить указатель на компоненту-функцию. Определить указатель на экземпляр класса. Написать демонстрационную программу, в которой создаются и разрушаются объекты пользовательского класса и каждый вызов конструктора и деструктора сопровождается выдачей соответствующего сообщения (какой объект какой конструктор или деструктор вызвал).

**Определение пользовательского класса:**

**Product.h**

#ifndef PRODUCT\_H

#define PRODUCT\_H

#include <iostream>

#include <string>

using namespace std;

class Product {

string name;

int amount;

float cost;

public:

Product(); //конструктор без параметров

Product(string name\_, int amount\_, float cost\_); //конструктор с параметрами

Product(const Product& prod); //конструктор копирования

~Product(); //деструктор

string get\_name() { //функция для доступа к полю name

return name;

}

int get\_amount() { //функция для доступа к полю amount

return amount;

}

float get\_cost() { //функция для доступа к полю cost

return cost;

}

void print(); //функция вывода содержимого полей экземпляра класса

void setName(string name\_); //функция для установки значения поля name

void setAmount(int amount\_); //функция для установки значения поля amount

void setCost(float cost\_); //функция для установки значения поля cost

};

#endif PRODUCT\_H

**Реализация конструктора и деструктора:**

**Product.cpp**

#include "Product.h"

Product::Product() : //реализация конструктора без параметров

name("Bread"), amount(3), cost(45.99)

{

cout << "Constructor without parameters called for Product: " << name << endl;

}

Product::Product(string name\_, int amount\_, float cost\_) : //реализация конструктора с параметрами

name(name\_), amount(amount\_), cost(cost\_)

{

cout << "Constructor with parameters called for Product: " << name << endl;

}

Product::~Product() { //реализация деструктора

cout << "Destructor called for Product: " << name << endl;

}

Product::Product(const Product&prod): //реализация конструктора копирования

name(prod.name), amount(prod.amount), cost(prod.cost)

{

cout << "Copying was there." << endl;

}

void Product::print() { //реализация функции вывода информации полей экземпляра класса в консоль

cout << "Name: " << name << "; Amount: " << amount << "; Cost: " << cost << endl;

}

void Product::setName(string name\_) { //реализация функции установки значения поля name

name = name\_;

}

void Product::setAmount(int amount\_) { //реализация функции установки значения поля amount

amount = amount\_;

}

void Product::setCost(float cost\_) { //реализация функции установки значения поля cost

cost = cost\_;

}

**Фрагмент программы, показывающий использование указателя на объект и указателя на функцию с объяснением:**

**Main()**

#include "Product.h"

int main() {

Product product\_1; //вызов конструктора без параметров

Product product\_2("Sugar", 12, 1200.45); //вызов конструктора с параметрами

Product product\_3(product\_1); //вызов конструктора копирования

Product\* product\_4 = new Product("Apple", 2, 540.72); //динамическое создание экземпляра класса с вызовом конструктора с параметрами

product\_1.print();

product\_4->setCost(999.99);

product\_4->print();

delete(product\_4); //вызов деструктора

Product set[3] = { Product("Juice", 11, 650), Product("Fish", 50, 900.19), Product("Tea", 9, 210.44) };

//Создание массива экземпляров класса

set[2].setAmount(39);

set[2].print();

void(Product:: \* pointer\_print)(); //создание указателя на функцию

pointer\_print = &Product::print;

(set[0].\*pointer\_print)();

Product\* pointer\_object = &product\_3; //создание указателя на экземпляр класса

cout << pointer\_object->get\_name() << endl;

return 0;

}

**Вывод:** В ходе лабораторной работы освоил основные навыки реализации классов на С++.