Министерство образования Республики Беларусь

УО «Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №2

По дисциплине: “Языки программирования”

Тема: “Наследование и виртуальные функции”Вариант №6

Выполнил: студент 2 курса

группы ПО-7

Комиссаров Андрей

Проверила: Дряпко А. В.

Брест 2021

Цель: получить практические навыки создания иерархии классов и использования статических компонентов класса.

Постановка задачи: написать программу, в которой создается иерархия классов. Включить полиморфные объекты в связанный список, используя статические компоненты класса. Показать использование виртуальных функций.

Порядок выполнения работы:

1. Определить иерархию классов (в соответствии с вариантом).
2. Определить в классе статическую компоненту - указатель на начало связанного списка объектов и статическую функцию для просмотра списка.
3. Реализовать классы.
4. Написать демонстрационную программу, в которой создаются
5. объекты различных классов и помещаются в список, после чего список просматривается.
6. Сделать соответствующие методы не виртуальными и посмотреть, что будет.
7. Реализовать вариант, когда объект добавляется в список при создании, т.е. в конструкторе (смотри пункт 6 следующего раздела).

Вариант задания: 6) журнал, книга, печатное издание, учебник;

Иерархия классов:

class PrintItem  
 const char \*name;  
 float price;

class Magazine: PrintItem  
 const char\* Publisher;

class Book : PrintItem  
 const char\* Author;

class TextBook : Book  
 int Grade;

**Код программы:**

**Содержание файла Main.CPP**

#include <iomanip>

#include "books.h"

int PrintItem::count = 0; PrintItem\*\* PrintItem::items = nullptr;

int main() {

setlocale(LC\_ALL, "Russian"); //русский язык

Magazine SportsMag("NBA Stars 2011", 9.99, "NBA Publishing ltd.");

SportsMag.Add(); //добавление первого элемента в список, остальные будут добавлятся при создании

Book CnP("Преступление и наказание", 10.59, "Ф. Достоевский"); //создание книги

Book FnC("Отцы и дети", 5.69, "И. Тургенев"); //создание книги

TextBook Math3("Математика", 3.33, "Министерство образования РБ", 3); //создание журнала

PrintItem::listItems(); //вывод элементов

}

**Содержание файла Books.CPP**

#include <iostream>

#include "books.h"

using namespace std;

PrintItem::PrintItem(const char\* name, float price, bool isAdd) :

name(name), price(price) {

if (isAdd) Add();

cout << "Создан PrintItem \"" << name << "\"\n";

}

PrintItem::~PrintItem() { cout << "Удалён PrintItem \"" << name << "\"\n"; }

void PrintItem::Add() {

PrintItem\*\* temp = items;

items = new PrintItem \* [count + 1];

for (int i = 0; i < count; i++) items[i] = temp[i];

items[count] = this; count++;

}

void PrintItem::listItems() {

cout << "\n=========================================\n";

for (int i = 0; i < count; i++) items[i]->Print();

cout << "=========================================\n" << endl;

}

Magazine::Magazine(const char\* name, float price, const char\* publisher) : PrintItem(name, price, false), publisher(publisher) { cout << "Создан журнал \"" << name << "\"\n"; }

Magazine::~Magazine() { cout << "Удалён журнал \"" << name << "\"\n"; }

void Magazine::Print() { cout << "Журнал с названием: \"" << name << "\", ценой: " << price << ", издателем: " << publisher << "\n"; }

Book::Book(const char\* name, float price, const char\* author) : PrintItem(name, price, true), author(author) { cout << "Создана книга \"" << name << "\"\n"; }

Book::~Book() { cout << "Удалена книга \"" << name << "\"\n"; }

void Book::Print() { cout << "Книга с названием: \"" << name << "\", ценой: " << price << ", автором: " << author << "\n"; }

TextBook::TextBook(const char\* name, float price, const char\* author, int grade) : Book(name, price, author), grade(grade) { cout << "Создан учебник \"" << name << "\"\n"; }

TextBook::~TextBook() { cout << "Удалён учебник \"" << name << "\"\n"; }

void TextBook::Print() { cout << "Учебник с названием: \"" << name << "\", ценой: " << price << ", автором: " << author << ", класс: " << grade << "\n"; }

**Содержание файла Books.H**

#ifndef BOOKS\_H

#define BOOKS\_H

class PrintItem {

protected:

const char\* name;

float price;

public:

static int count;

static PrintItem\*\* items;

PrintItem(const char\* name, float price, bool isAdd = true);

~PrintItem();

virtual void Print() = 0;

virtual void Add() final;

static void listItems();

};

class Magazine : public PrintItem {

protected: const char\* publisher;

public:

Magazine(const char\* name, float price, const char\* publisher);

~Magazine();

void Print() override;

};

class Book : protected PrintItem {

protected: const char\* author;

public:

Book(const char\* name, float price, const char\* author);

~Book();

void Print() override;

};

class TextBook : Book {

protected: int grade;

public:

TextBook(const char\* name, float price, const char\* author, int grade);

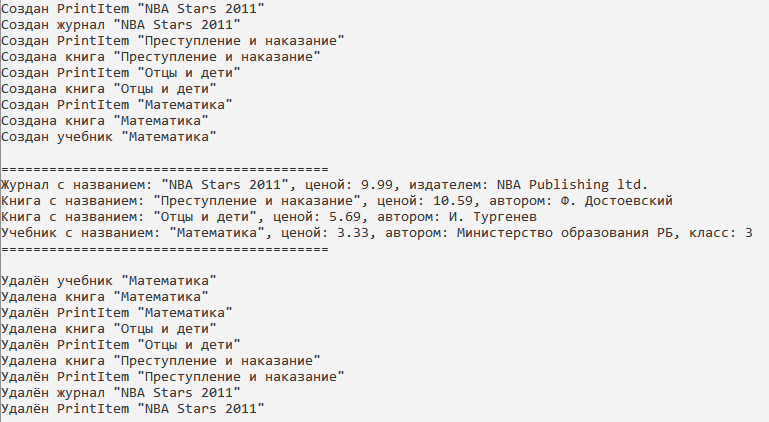
~TextBook();

void Print() override;

};

#endif

**Результат работы программы:**

****

**Вывод:** Я получил практические навыки создания иерархии классов и использования статических компонентов класса.