Министерство образования Республики Беларусь

Учреждение образования

БелорусскиЙ государственный университет

информатики и радиоэлектроники

Факультет инженерно-экономический

Кафедра экономической информатики

**Отчет к лабораторной работе №4**

**по теме**

**«Потоки ввода/вывода»**

|  |  |
| --- | --- |
| Выполнил: | Вдовенко Н.Д.  студент группы 272303 |
| Проверила: | Салапура М. Н.  ст. преподаватель |

Минск 2023

Общая постановка задачи

**Цель работы:** изучить особенности потоков ввода/вывода на языке С++.

**Индивидуальное задание:**

2.На основе иерархии классов, разработанной в лабораторной работе «ОБРАБОТКА ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ В ЯЗЫКЕ С++», реализовать программу для работы с данными, используя потоки ввода- вывода. Реализовать функций просмотра, добавления, удаления, редактирования, сортировки, поиска данных по необходимым параметрам. Использовать минимум 5 флагов форматирования. Сформировать табличный вывод всех данных. Использовать функции setf() и unsetf() для установки и сбрасывания флагов. Использовать функции eof(), fail(), good(), bad(), clear() для проверки состояния потока.

**Краткие теоретические сведения:**

Объект cout представляет стандартный выходной поток и по умолчанию выводит данные на экран. Объект cin представляет стандартный входной поток и по умолчанию считывает данные с клавиатуры

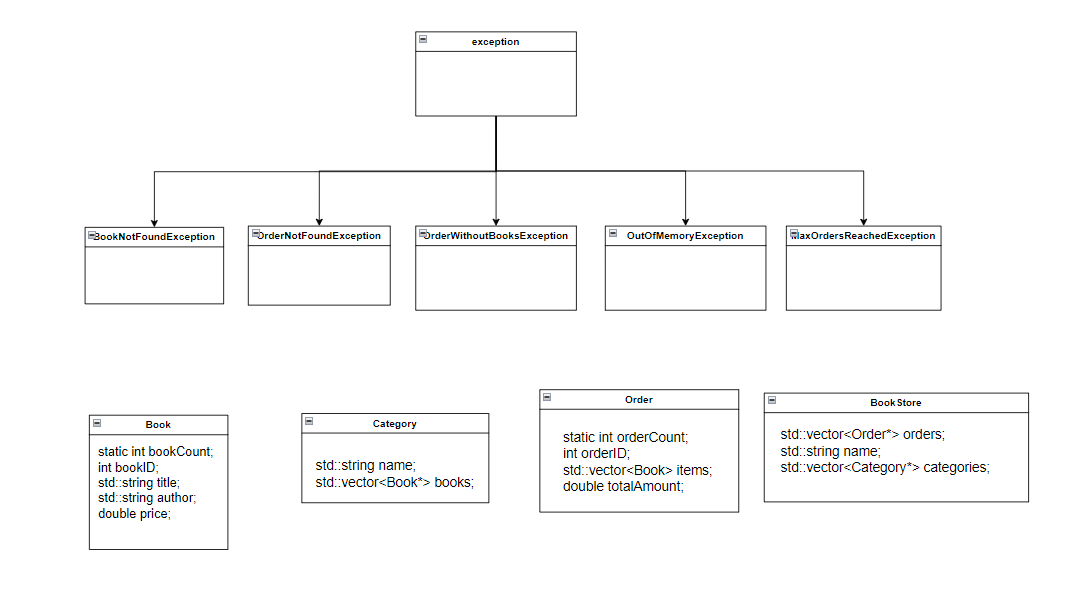
Манипуляторы устанавливают в объектах cin и cout параметры текущего оператора ввода-вывода.

Предопределенные потоковые объекты cin и cout по умолчанию связаны с клавиатурой и монитором соответственно. Объект cin определяет глобальный поток ввода и управляет извлечением из стандартного входного потока в виде потока байтов. Объект cout определяет глобальный поток вывода и управляет вставками в стандартный вывод в виде потока байтов.

У любого потока есть набор флагов, с помощью которых можно следить за состоянием потока, т.е. определять есть ли в нем ошибки.

Символы, в определенных ситуациях не несущие смысловой нагрузки, (пробелы, '\n', табуляция) могут пропускаться при вводе данных. Это может привести к некоторым нежелательным побочным эффектам. Важно иметь возможность дать команду входному потоку игнорировать или не игнорировать разделители. Это делается с помощью флага skipws.

**Диаграмма классов:**

****

**Контрольные вопросы:**

1. Поток в C++ — это последовательность данных, которая может быть введена или выведена из программы. Потоки представляют собой интерфейс для взаимодействия с вводом и выводом данных.

2. iostream — это базовый класс для большинства потоковых классов в C++. Включает в себя функциональность для стандартного ввода и вывода через cin и cout.

3. Да, это истинное утверждение. Потоки могут быть входными (например, cin для ввода данных) или выходными (например, cout для вывода данных).

4. Флаг skipws определяет, игнорировать ли пробельные символы при вводе данных с использованием cin. Если этот флаг установлен (через cin >> skipws;), то пробелы и символы табуляции будут проигнорированы при вводе данных.

5. int main(int argc, char\* argv[])

6. Стандартные потоки ввода и вывода предоставляют интерфейс для взаимодействия программы с пользователем через консоль. cin используется для ввода данных, а cout для вывода данных.

7. Для работы с потоками ввода/вывода в C++ необходимо подключить заголовочный файл <iostream>.

8. Поток форматированного вывода ostream позволяет форматировать данные, которые будут выведены. Функции, такие как setw, setfill и другие функции из заголовочного файла <iomanip>, обеспечивают форматированный вывод.

9. Библиотека iomanip предназначена для форматирования вывода данных. Она предоставляет функции для управления шириной поля, точностью вывода чисел с плавающей запятой и другие возможности форматирования вывода.

10. string input;

getline(cin, input);

**Листинг кода:**

#include "libs.h"

#include "InputChecks.h"

#include "classes.h"

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

try {

BookStore myBookstore("My Bookstore");

Category fiction("Художественная литература");

fiction.AddBook(new Book("Война и мир", "Лев Толстой", 85.50));

fiction.AddBook(new Book("Преступление и наказание", "Федор Достоевский", 45.25));

fiction.AddBook(new Book("Мастер и Маргарита", "Михаил Булгаков", 32.00));

fiction.AddBook(new Book("1984", "Джордж Оруэлл", 20.50));

fiction.AddBook(new Book("Три товарища", "Эрих Мария Ремарк", 38.75));

Category nonFiction("Научно-популярная литература");

nonFiction.AddBook(new Book("Краткая история времени", "Стивен Хокинг", 28.75));

nonFiction.AddBook(new Book("Человек и его символы", "Карл Юнг", 95.00));

nonFiction.AddBook(new Book("Космос: Частная история", "Карл Саган", 42.30));

nonFiction.AddBook(new Book("Биология человека", "Джон Джонсон", 55.90));

Category biography("Биографии и мемуары");

biography.AddBook(new Book("Стив Джобс: Биография", "Уолтер Айзексон", 42.75));

biography.AddBook(new Book("Мой дедушка", "Артур Шопенгауэр", 18.90));

myBookstore.AddCategory(&fiction);

myBookstore.AddCategory(&nonFiction);

myBookstore.AddCategory(&biography);

int choice = 0;

while (true) {

cout << "\nMenu options" << endl;

cout << "1 - Place an order" << endl;

cout << "2 - Show orders" << endl;

cout << "3 - Find an order" << endl;

cout << "4 - Delete an order" << endl;

cout << "5 - Change an order" << endl;

cout << "6 - Sort orders by price" << endl;

cout << "7 - Sort books by price" << endl;

cout << "0 - Exit" << endl;

cout << "Choose an action: ";

choice = GetIntNumber();

switch (choice) {

case 1: {

myBookstore.PlaceOrder();

break;

}

case 2: {

myBookstore.ShowOrders();

break;

}

case 3: {

myBookstore.FindOrder();

break;

}

case 4: {

myBookstore.DeleteOrder();

break;

}

case 5: {

myBookstore.ChangeOrder();

break;

}

case 6: {

myBookstore.SortOrdersByPrice();

break;

}

case 7: {

myBookstore.SortBooksByPrice();

break;

}

case 0: {

std::cout << "Exiting the bookstore. Goodbye!" << std::endl;

return 0;

}

default: {

std::cerr << "Invalid choice. Please choose a valid option." << std::endl;

break;

}

}

}

}

catch (exception& ex) {

std::cerr << "Exception: " << ex.what() << std::endl;

}

return 0;

}



**Вывод**: в ходе выполнения данной лабораторной работы по теме "Потоки ввода/вывода" были изучены и применены основные принципы работы с потоками в программировании.