|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет  имени Н. Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н. Э. Баумана)** |
| ФАКУЛЬТЕТ | «Информатика, искусственный интеллект и системы управления» (ИУ) |

|  |  |
| --- | --- |
| КАФЕДРА | «Информационная безопасность» (ИУ8) |

Лабораторная работа № 4

ПО КУРСУ

«Алгоритмические языки»

на тему «Использование объектов своих классов в последовательных контейнерах библиотеки STL »

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент | ИУ8-24 |  |  |  | Кулиев О.М. |
|  | (Группа) |  |  |  | (И. О. Фамилия) |
|  |  |  |  |  |  |
| Преподаватель: |  |  |  |  | Барыкин Д. В |
|  |  |  |  |  | (И.О. Фамилия) |

**Цель работы:**

Наработка навыков по использованию объектов своих классов в последовательных контейнерах библиотеки STL.

**Текст программы:**

#include <iostream>

#include<vector>

#include <iostream>

#include <string>

#include <algorithm>

#include <fstream>

#include<deque>

#include<sstream>

using namespace std;

class kniga {

public:

string name;

string god;

string nazv;

string izdat;

int chist;

kniga() {}

kniga(string n, string g, string naz, string iz, int ch) :name(n), god(g), nazv(naz), izdat(iz), chist(ch) {}

kniga(kniga& copy) {

this->name = copy.name;

this->god = copy.god;

this->nazv = copy.nazv;

this->izdat = copy.izdat;

this->chist = copy.chist;

}

kniga(kniga&& perem) {

swap(this->name, perem.name);

swap(this->god, perem.god);

swap(this->nazv, perem.nazv);

swap(this->izdat, perem.izdat);

swap(this->chist, perem.chist);

}

kniga& operator=(const kniga& k1) {

this->name = k1.name;

this->god = k1.god;

this->nazv = k1.nazv;

this->izdat = k1.izdat;

this->chist = k1.chist;

}

friend ostream& operator<<(ostream& o, const kniga& k) {

cout << k.name << ": " << k.god << k.nazv << "," << k.izdat << "," << k.chist << endl;

return o;

}

};

bool comp(const kniga& k1, const kniga& k2) {

return (k1.name < k2.name);

}

int main()

{

vector<kniga> spisok;

ifstream input("input.txt");

string name, nazv, izdat, god;int chist;

while (getline(input, name)) {

getline(input, god);

input >> nazv >> izdat >> chist;

input.ignore();

spisok.push\_back(kniga(name,god, nazv, izdat, chist));

}

ofstream output("output.txt");

output << "Original container:" << endl;

cout << "Original container:" << endl;

for (const auto& x : spisok) {

output << x << endl;

}

deque<kniga> copied\_spisok;

copied\_spisok.resize(spisok.size());

copy(spisok.begin(), spisok.end(), copied\_spisok.begin());

sort(spisok.begin(), spisok.end(), comp);

output << "Sorted container:" << endl;

cout << "Sorted container:" << endl;

for (const auto& x : spisok) {

output << x << endl;

}

output << "Copied container:" << endl;

cout << "Copied container:" << endl;

for (const auto& x : copied\_spisok) {

output << x << endl;

}

}

**Вывод:**

Выполнив лабораторную работу №4, я изучил тему «Использование объектов своих классов в последовательных контейнерах библиотеки STL»понял как решать такого рода примеры и улучшил свои навыки в этой теме.