|  |  |
| --- | --- |
|  | **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

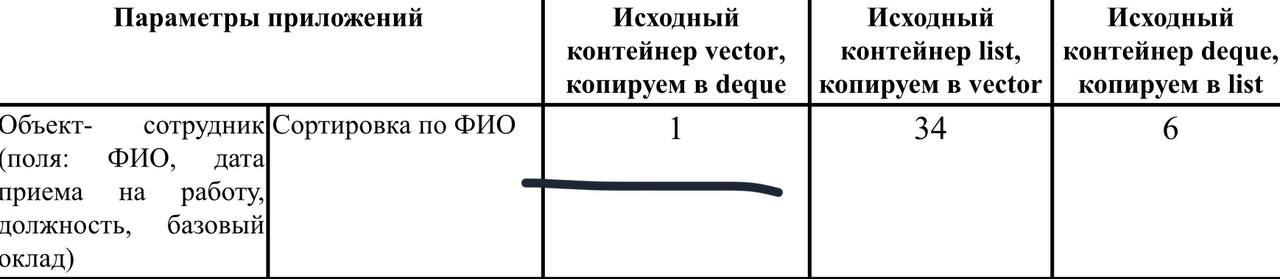
«Алгоритмические языки»

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент | ИУ8-24 |  |  |  | Алексеев К. А. |
|  | (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 <sstream>

#include <fstream>

#include <vector>

#include <algorithm>

#include <deque>

using namespace std;

class Worker {

public:

string FIO;

string Joined;

string Speciality;

int Salary;

Worker() {}

Worker(string a, string b, string c, int d) : FIO(a), Joined(b), Speciality(c), Salary(d) {}

Worker(Worker& copy) {

this->FIO = copy.FIO;

this->Joined = copy.Joined;

this->Speciality = copy.Speciality;

this->Salary = copy.Salary;

}

Worker(Worker&& perem) {

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

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

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

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

}

Worker& operator=(const Worker& w1) {

this->FIO = w1.FIO;

this->Joined = w1.Joined;

this->Speciality = w1.Speciality;

this->Salary = w1.Salary;

return \*this;

}

friend ostream& operator<<(ostream& os, const Worker& w) {

cout << w.FIO << ": " << w.Joined << "," << w.Speciality << "," << w.Salary << endl;

return os;

}

};

bool compare(const Worker& w1, const Worker& w2) {

return (w1.FIO < w2.FIO);

}

int main() {

vector<Worker> spisok;

ifstream input("input.txt");

string FIO, Joined, Speciality; int Salary;

while (getline(input, FIO)) {

getline(input, Joined);

input >> Speciality >> Salary;

input.ignore();

spisok.push\_back(Worker(FIO, Joined, Speciality, Salary));

}

ofstream output("output.txt");

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

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

for (const auto& x : spisok) {

output << x << endl;

}

deque<Worker> copied\_spisok;

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

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

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

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) {

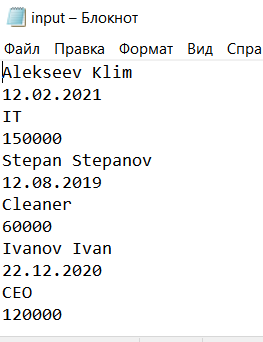
output << x << endl;

}

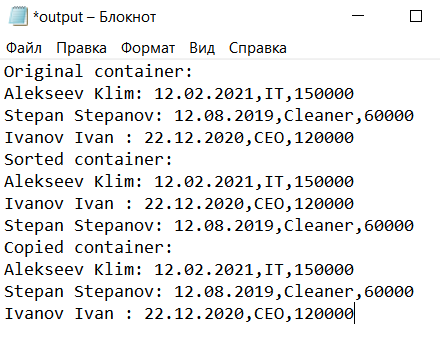
return 0;

}

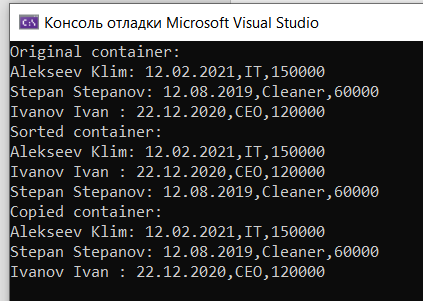
**Файл input.txt:**



**Файл output.txt:**



**Консоль:**



**Вывод:**

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