Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Пермский национальный исследовательский**

**политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Ё Т**

**по лабораторной работе №8**

Дисциплина: основы алгоритмизации и программирования

Тема: “Структуры данных”

**Вар.3**

Выполнил работу:

студент группы ИВТ-20-2Б

Солдатов А.П.

Проверил:

Доцент кафедры ИТАС

Полякова О.А.

Пермь, 2020

**Цель задачи**

Цель – написать программу с использованием структур данных и

выполнение операций с ними.

Для достижения поставленной цели необходимо решить следующие задачи:

* Понять, как создаются и объявляются структуры данных.
* Понять, как работают структуры данных.
* Понять, оптимальные методы ввода информации в структуру

данных.

* Разработать программу по найденному материалу.

**Постановка задачи**

Структура "Государство":

- название;

- столица;

- численность населения;

- занимаемая площадь.

Удалить все элементы, у которых численность меньше заданной, добавить элемент после элемента с указанным номером.

**Анализ задачи**

1. Определить какие операции должны быть выполнены по заданию:

* “Отсеивание” элементов, опираясь на их численность населения
* Добавление элемента после элемента с заданным номером
* Для названия страны и её столицы созданы 2 поля типа string

Для числености населения и площади страны созданы 2 поля типа long int

**2.** Создание структуры “Country” и её объявление:

struct Country

{

string name;

string capital;

long int population;

long int square;

};

3 . Создание массива структур:

Country N1 = { "Россия", "Москва", 146748590, 17098246 };

Country N2 = { "Германия", "Берлин", 83149300, 357578 };

Country N3 = { "США", "Вашингтон", 328239523, 9826675 };

Country N4 = { "Китай", "Пекин", 1439323776, 9596961 };

Country a[4] = { N4, N1, N2, N3 };

4. Вывод данных осуществляется через функцию:

cout << " " << i << ") " << '\t' << "Название:\t" << a[i].name << endl << '\t' << "Столица:\t" << a[i].capital << endl << '\t' << "Население:\t" << a[i].population << endl << '\t' << "Площщадь:\t" << a[i].square << endl << endl;

5.Для решения задачи (добавление элемента после элемента с заданным номером, “отсеивание” элементов, у которых численность населения меньше заданной) используются циклы и ветвления. Например:

if (i != numbercheck && a[i].population > populationcheck)

cout << " " << '\t' << "Название:\t" << a[i].name << endl << '\t' << "Столица:\t" << a[i].capital << endl << '\t' << "Население:\t" << a[i].population << endl << '\t' << "Площщадь:\t" << a[i].square << endl << endl;

numbercheck - переменная типа long int, которая содержит в себе номер элемента, после которого нужно добавить новый

populationcheck - переменная типа long int, которая содержит в себе минимально возможную численность населения.

**Код**

#include <iostream>

#include <string>

using namespace std;

struct Country

{

string name;

string capital;

long int population;

long int square;

};

int main()

{

system("chcp 1251>nul");

int i, numbercheck;

long int populationcheck;

Country N1 = { "Россия", "Москва", 146748590, 17098246 };

Country N2 = { "Германия", "Берлин", 83149300, 357578 };

Country N3 = { "США", "Вашингтон", 328239523, 9826675 };

Country N4 = { "Китай", "Пекин", 1439323776, 9596961 };

Country a[4] = { N4, N1, N2, N3 };

for (i = 1; i < 4; i++)

cout << " " << i << ") " << '\t' << "Название:\t" << a[i].name << endl << '\t' << "Столица:\t" << a[i].capital << endl << '\t' << "Население:\t" << a[i].population << endl << '\t' << "Площщадь:\t" << a[i].square << endl << endl;

do

{

cout << "Введи минимальную численость населения "; cin >> populationcheck;

} while (populationcheck < 1);

do

{

cout << "Введи номер, после которого добавить элемент: "; cin >> numbercheck;

} while (numbercheck < 1 || numbercheck > 3);

cout << "Изменения:" << endl;

i = 1;

for (i = 1; i < 4; i++)

if (i != numbercheck && a[i].population > populationcheck)

cout << " " << '\t' << "Название:\t" << a[i].name << endl << '\t' << "Столица:\t" << a[i].capital << endl << '\t' << "Население:\t" << a[i].population << endl << '\t' << "Площщадь:\t" << a[i].square << endl << endl;

else if (i == numbercheck)

{

if (a[i].population < populationcheck)

cout << " " << '\t' << "Название:\t" << a[0].name << endl << '\t' << "Столица:\t" << a[0].capital << endl << '\t' << "Население:\t" << a[0].population << endl << '\t' << "Площщадь:\t" << a[0].square << endl << endl;

else

{

cout << " " << '\t' << "Название:\t" << a[i].name << endl << '\t' << "Столица:\t" << a[i].capital << endl << '\t' << "Население:\t" << a[i].population << endl << '\t' << "Площщадь:\t" << a[i].square << endl << endl;

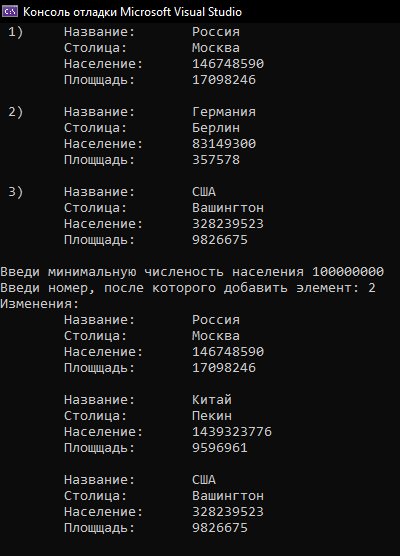
cout << " " << '\t' << "Название:\t" << a[0].name << endl << '\t' << "Столица:\t" << a[0].capital << endl << '\t' << "Население:\t" << a[0].population << endl << '\t' << "Площщадь:\t" << a[0].square << endl << endl;

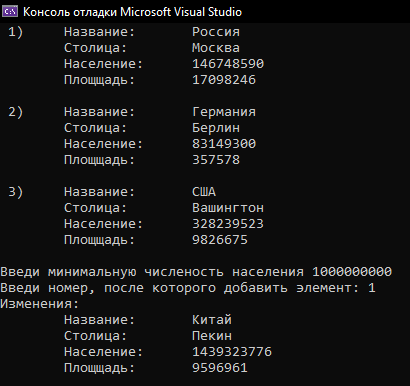
}

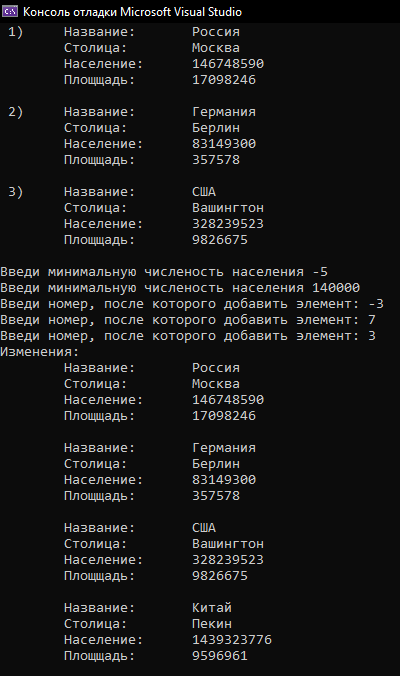
}

}

**Работа кода**





**Блок-схема**

