Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №2

По дисциплине «Основы алгоритмизации и программирования»

За 2 семестр

Тема: «Структуры, объединения, перечисления»

Выполнили:

Студенты 1 курса

Группы ИИ-21(1)

Кабак Д. Н.

Проверил:

Гирель Т. Н.

Брест 2022

**Цель:** научиться работать со структурными переменными, изучить особенности различных видов структур, а также сферы их применения.

**Ход работы**

Задание:

Реализовать структуру по заданию:

*3. Структура «Государство»: название; столица; численность населения; занимаемая*

*площадь.*

Код:

#include<iostream>

#include <string>

using namespace std;

struct State {

    string stateName;

    string capital;

    string population;

    string area;

};

State\* AddStruct(State\* Obj, const int amount);

void setData(State\* Obj, const int amount);

void sortStruct(State\* Obj, const int amount);

void search(State\* Obj, const int amount, string str);

void editData(State\* Obj, const int amount, int str);

State\* DeleteStruct(State\* Obj, const int amount, int strDelete);

void showData(const State\* Obj, const int amount);

int main() {

    system("chcp 1251 > nul");

    State\* country = 0;

    int countryAmount = 0;

    int YesOrNot = 0;

    while (true) {

        cout << "1. Ввод массива структур.\n";

        cout << "2. Сортировка массива структур по заданному параметру.\n";

        cout << "3. Поиск в массиве структур по заданному параметру.\n";

        cout << "4. Изменение заданной структуры.\n";

        cout << "5. Удаление структуры из массива.\n";

        cout << "6. Вывод на экран массива структур.\n";

        cout << "7. Выход.\n";

        int n;

        cout << "\nВведите число (ваш выбор): ";

        cin >> n;

        cin.get();

        system("cls");

        switch (n) {

        case 1: { //ввод

            do {

                country = AddStruct(country, countryAmount);

                setData(country, countryAmount);

                countryAmount++;

                cout << "Продолжить ввод данных (1 - да, 0 - нет): ";

                cin >> YesOrNot;

                cin.get();

            } while (YesOrNot != 0);

            system("cls");

        }

            break;

        case 2: // сортировка

            sortStruct(country, countryAmount);

            showData(country, countryAmount);

            break;

        case 3: { // поиск

            string str;

            cout << "Поиск: ";

            getline(cin, str);

            search(country, countryAmount, str);

        }

            break;

        case 4: { // изменение

            showData(country, countryAmount);

            cout << "Введите номер строки, которую хотите изменить: " << endl;

            int str;

            cin >> str;

            cin.get();

            cout << "1. Государство.\n";

            cout << "2. Столица.\n";

            cout << "3. Население.\n";

            cout << "4. Площадь.\n";

            cout << "Введите номер атрибута, который хотите изменить: " << endl;

            int atr;

            cin >> atr;

            cin.get();

            cout << "Теперь введите нужное вам значение:" << endl;

            editData(country, str, atr);

        }

            break;

        case 5: { // Удаление

            showData(country, countryAmount);

            cout << "\nВведите номер строки, которую хотите удалить: " << endl;

            int str;

            cin >> str;

            cin.get();

            country = DeleteStruct(country, countryAmount, str);

            countryAmount--;

        }

            break;

        case 6: // вывод

            showData(country, countryAmount);

            break;

        case 7: // выход

            exit(0);

            break;

        default:

            cout << "Такого пункта в меню нет. Попробуйте ещё. ";

            break;

        }

    }

    delete[] country;

    return 0;

}

State\* AddStruct(State\* Obj, const int amount) {

    if (amount == 0) {

        Obj = new State[amount + 1];

    }

    else {

        State\* tempObj = new State[amount + 1];

        for (int i = 0; i < amount; i++) {

            tempObj[i] = Obj[i];

        }

        delete[] Obj;

        Obj = tempObj;

    }

    return Obj;

}

void setData(State\* Obj, const int amount) {

    cout << "Государство: ";

    getline(cin, Obj[amount].stateName);

    cout << "Столица: ";

    getline(cin, Obj[amount].capital);

    cout << "Население: ";

    getline(cin, Obj[amount].population);

    cout << "Площадь: ";

    getline(cin, Obj[amount].area);

    cout << endl;

}

void sortStruct(State\* Obj, const int amount) {

    cout << "1. Государство.\n";

    cout << "2. Столица.\n";

    cout << "3. Население.\n";

    cout << "4. Площадь.\n";

    int atr;

    cin >> atr;

    cin.get();

    State buffer;

    switch (atr) {

    case 1:

        for (int i = 0; i < amount - 1; i++) {

            for (int j = 0; j < amount - i - 1; j++) {

                if (Obj[j].stateName > Obj[j + 1].stateName) {

                    buffer = Obj[j];

                    Obj[j] = Obj[j + 1];

                    Obj[j + 1] = buffer;

                }

            }

        }

    break;

    case 2:

        for (int i = 0; i < amount - 1; i++) {

            for (int j = 0; j < amount - i - 1; j++) {

                if (Obj[j].capital > Obj[j + 1].capital) {

                    buffer = Obj[j];

                    Obj[j] = Obj[j + 1];

                    Obj[j + 1] = buffer;

                }

            }

        }

    break;

    case 3:

        for (int i = 0; i < amount - 1; i++) {

            for (int j = 0; j < amount - i - 1; j++) {

                if (Obj[j].population > Obj[j + 1].population) {

                    buffer = Obj[j];

                    Obj[j] = Obj[j + 1];

                    Obj[j + 1] = buffer;

                }

            }

        }

    break;

    case 4:

        for (int i = 0; i < amount - 1; i++) {

            for (int j = 0; j < amount - i - 1; j++) {

                if (Obj[j].area > Obj[j + 1].area) {

                    buffer = Obj[j];

                    Obj[j] = Obj[j + 1];

                    Obj[j + 1] = buffer;

                }

            }

        }

        break;

    default:

        cout << "Такого пункта в меню нет! ";

        break;

    }

}

void search(State\* Obj, const int amount, string str) {

    int k = 0;

    for (int i = 0; i < amount; i++) {

        if (str == Obj[i].stateName || str == Obj[i].capital || str == Obj[i].population || str == Obj[i].area) {

            cout << left << setw(10) << "№  " << setw(10) << "Государство\t" << setw(10) << "Столица\t"

             << setw(10) << "Населениеt" << setw(10) << "Площадь\t" << endl;

            cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" << endl;

            cout << left << setw(10) << i + 1 << setw(10) << Obj[i].stateName << setw(10) << Obj[i].capital

             << setw(10) << Obj[i].area << setw(10) <<  Obj[i].population << endl;

        }

        else k += 1;

    }

    if (k == amount) cout << "Результат не найден.\n" << endl;

}

void editData(State\* Obj, const int amount, int atr) {

    switch (atr) {

    case 1:

        cout << "Государство: ";

        getline(cin, \*&Obj[amount-1].stateName);

        break;

    case 2:

        cout << "Столица: ";

        getline(cin, \*&Obj[amount-1].capital);

        break;

    case 3:

        cout << "Население: ";

        getline(cin, \*&Obj[amount-1].population);

        break;

    case 4:

        cout << "Площадь: ";

        getline(cin, \*&Obj[amount-1].area);

        break;

    default:

        cout << "Такого пункта в меню нет! ";

        break;

    }

    cout << endl;

}

State\* DeleteStruct(State\* Obj, const int amount, int str) {

    if (amount == 0) {

        cout << "ERROR! Вы еще не создали хотя бы одну структуру!";

    }

    if (amount == 1) delete[] Obj;

    if (amount > 1) {

        State\* tempObj = new State[amount - 1];

        for (int i = 0; i < amount - 1; i++) {

            if (i < str - 1) tempObj[i] = Obj[i];

            else tempObj[i] = Obj[i + 1];

        }

        delete[] Obj;

        Obj = tempObj;

    }

    return Obj;

}

void showData(const State\* Obj, const int amount) {

    system("cls");

    cout << left << setw(10) << "№  " << setw(10) << "Государство\t" << setw(10) << "Столица\t"

     << setw(10) << "Населениеt" << setw(10) << "Площадь\t" << endl;

    cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" << endl;

    for (int i = 0; i < amount; i++) {

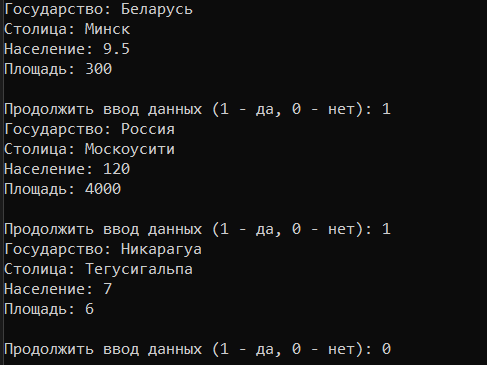
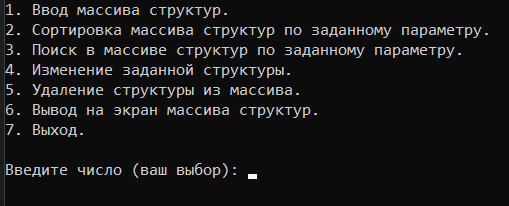
        cout << left << setw(10) << i + 1 << setw(10) << Obj[i].stateName << setw(10) << Obj[i].capital

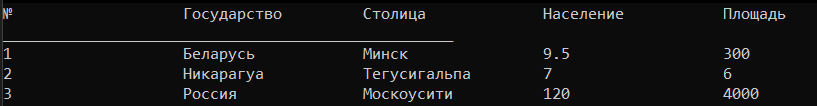
         << setw(10) << Obj[i].area << setw(10) <<  Obj[i].population << endl;

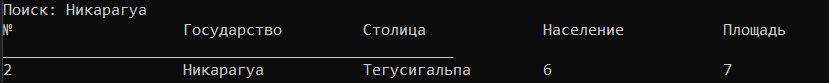
    }

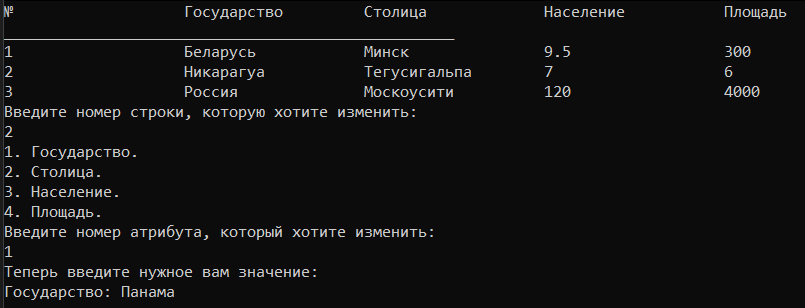
}

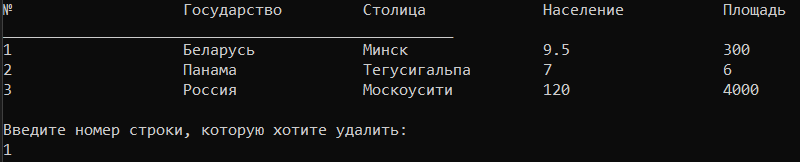
Вывод:











**Вывод:** научился работать со структурными переменными, изучил особенности различных видов структур, а также сферы их применения.