Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

Лабораторная работа 6

По дисциплине «Структуры данных»

На тему «Редактирование файлов: изменение, удаление поля»

Выполнила:

Студентка 1 курса 2 группы

Глухова Д.В.

Преподаватель: Север А.С.

2023, Минск

Структуры часто образуют массивы. Чтобы объявить массив структур, вначале необходимо определить структуру (то есть определить агрегатный тип данных), а затем объявить переменную массива этого же типа. Для начала рассмотрим как объявить структуру:

Поскольку структуры определяются программистом, то вначале мы должны сообщить компилятору, как она вообще будет выглядеть. Для этого используется **ключевое слово** struct. Переменные, которые являются частью структуры, называются **членами** **структуры** (или ***«полями структуры»***).

***Предупреждение:*** Одна из самых простых ошибок в C++ — забыть точку с запятой в конце объявления структуры. Это приведет к ошибке компиляции в следующей строке кода. Современные компиляторы, такие как Visual Studio версии 2010, а также более новых версий, укажут вам, что вы забыли точку с запятой в конце, но более старые компиляторы могут этого и не сделать, из-за чего такую ошибку будет трудно найти.

## Доступ к членам структур

Когда мы объявляем переменную структуры, например, Employee john, то john ссылается на всю структуру. Для того, чтобы получить доступ к отдельным её членам, используется **оператор выбора члена** (.).

**Вложенные структуры**

Одни структуры могут содержать другие структуры. Например:

struct Employee

{

short id;

int age;

double salary;

};

struct Company

{

Employee CEO; // Employee - это структура внутри структуры Company

int numberOfEmployees;

};

Company myCompany;

**Функция полного удаления:**

Для того, чтобы полностью удалить данные нам не нужна новая функция, для этого достаточно в main очистить динамический массив с помощью delete. И создать новый динамический массив.

**Задание для лабораторной работы №6:**

Написать функции удаления (выборочное и полное) и изменения массива структур для вашей БД (из лабораторной работы №2).

Задание из лабораторной работы №2

Вы являетесь сотрудником коммерческого отдела компании, продающей различные товары через Интернет. Вашей задачей является отслеживание финансовой составляющей работы компании.  
Работа Вашей компании организована следующим образом: на Интернет-сайте компании представлены (выставлены на продажу) некоторые товары. Каждый из них имеет некоторое название, цену и единицу измерения (штуки, килограммы, литры).  
Для проведения исследований и оптимизации работы магазина Вы пытаетесь собирать данные с Ваших клиентов. При этом для Вас определяющее значение имеют стандартные анкетные данные, а также телефон и адрес электронной почты для связи.  
По каждому факту продажи Вы автоматически фиксируете клиента, товары, количество, дату продажи, дату доставки.

|  |
| --- |
| Код программы |
| #include <iostream>  #include <fstream>  #include <string>  #include <vector>  #include <windows.h>  # define str\_len 40  # define SIZE 40  using namespace std;  struct Initial {  char fio[str\_len];  char telephone[str\_len];  char email[str\_len];  char product[str\_len];  int numProduct;  char dateSale[str\_len];  char dateDelivery[str\_len];  };  vector<Initial> initial;  struct Initial bad;  void ReadingData(string fileName)  {  cout << "=======================================\n";  for (auto client : initial)  {  cout << "Фамилия клиента: " << client.fio << "\n";  cout << "Почта клиента: " << client.email << "\n";  cout << "Телефон клиента: " << client.telephone << "\n";  cout << "Купленный товар: " << client.product << "\n";  cout << "Количество товаров: " << client.numProduct << "\n";  cout << "Дата продажи: " << client.dateSale << "\n";  cout << "Дата доставки: " << client.dateDelivery << "\n";  cout << "=======================================\n";    }  }  void ReadFromFile(string fileName) {  ifstream read(fileName, ios::app);  if (read) {  while (!read.eof())  {  Initial temp;  read >> temp.fio;  read >> temp.email;  read >> temp.telephone;  read >> temp.product;  read >> temp.numProduct;  read >> temp.dateSale;  read >> temp.dateDelivery;  initial.push\_back(temp);  }  }  else  cout << "Ошибка открытия файла!" << endl;  read.close();  }  void WriteInFile(string fileName) {  ofstream record(fileName, ios::app);  if (record) {  for (auto client : initial)  {  record << client.fio << "\n";  record << client.email << "\n";  record << client.telephone << "\n";  record << client.product << "\n";  record << client.numProduct << "\n";  record << client.dateSale << "\n";  record << client.dateDelivery << "\n";  }  }  else  cout << "Ошибка открытия файла!" << endl;  record.close();  }  void SavingData(string fileName)  {  ofstream record(fileName, ios::app);  if (record) {  cout << "Ввод информации" << endl;  Initial list\_of\_clients;  cout << endl << "Фамилия клиента: " << endl;  cin >> list\_of\_clients.fio;  cout << "Номер телефона: " << endl;  cin >> list\_of\_clients.telephone;  cout << "Электронная почта: " << endl;  cin >> list\_of\_clients.email;  cout << "Товар: " << endl;  cin >> list\_of\_clients.product;  cout << "Количество товаров: " << endl;  cin >> list\_of\_clients.numProduct;  cout << "Дата продажи: " << endl;  cin >> list\_of\_clients.dateSale;  cout << "Дата доставки: " << endl;  cin >> list\_of\_clients.dateDelivery;  initial.push\_back(list\_of\_clients);  }  else  cout << "Ошибка открытия файла!" << endl;  record.close();  }  void Change()  {  int ind;  cout << "Input index for changing\n";  cin >> ind;  cout << endl << "Фамилия клиента: " << endl;  cin >> initial[ind].fio;  cout << "Номер телефона: " << endl;  cin >> initial[ind].telephone;  cout << "Электронная почта: " << endl;  cin >> initial[ind].email;  cout << "Товар: " << endl;  cin >> initial[ind].product;  cout << "Количество товаров: " << endl;  cin >> initial[ind].numProduct;  cout << "Дата продажи: " << endl;  cin >> initial[ind].dateSale;  cout << "Дата доставки: " << endl;  cin >> initial[ind].dateDelivery;  }  void delInFile(string fileName) {  ofstream file("Output.txt", ios\_base::trunc);  }  void DeleteData()  {  int d;  cout << "\n 1 - Удалить определённый фрагмент\n 2 - Удалить всё ";  cin >> d;  auto iter = initial.cbegin();  switch (d)  {  case 1:  int ind;  cout << "Выберете фрагмент от 0 до " << initial.size() << "\n";  cin >> ind;  initial.erase(ind + iter);  break;  case 2:  initial.clear();  break;  default:  break;  }  }  int main()  {  SetConsoleCP(1251);  SetConsoleOutputCP(1251);  setlocale(LC\_ALL, ".1251");  int filevar = NULL, var = NULL;  string fileName;  do {  cout << "ФАЙЛ: \n 1-ваше имя файла\n 2-по стандарту\n 0 - выход\n";  cin >> filevar;  switch (filevar)  {  case 1:  {  cout << "введите ваше имя файла\n";  cin >> fileName;  do {  cout << "\n1-вывод в консоль" << endl;  cout << "2-для ввода новой записи" << endl;  cout << "3-запись в файл" << endl;  cout << "4-чтение из файла" << endl;  cout << "5-удаление всего содержимого файла" << endl;  cout << "6-частичное удаление информации" << endl;  cout << "7-изменение информации" << endl;  cout << "0-для выхода" << endl;  cin >> var;  switch (var)  {  case 1: ReadingData(fileName); break;  case 2: SavingData(fileName); break;  case 3: WriteInFile(fileName); break;  case 4: ReadFromFile(fileName); break;  case 5: delInFile(fileName); break;  case 6: DeleteData(); break;  case 7: Change(); break;  default:  break;  }  } while (var != 0);  }  case 2:  {  do  {  cout << "\n1-вывод в консоль" << endl;  cout << "2-для ввода новой записи" << endl;  cout << "3-запись в файл" << endl;  cout << "4-чтение из файла" << endl;  cout << "5-удаление всего содержимого файла" << endl;  cout << "6-частичное удаление информации" << endl;  cout << "7-изменение информации" << endl;  cout << "0-для выхода" << endl;  cin >> var;  switch (var)  {  case 1: ReadingData("Output.txt"); break;  case 2: SavingData("Output.txt"); break;  case 3: WriteInFile("Output.txt"); break;  case 4: ReadFromFile("Output.txt"); break;  case 5: delInFile("Output.txt"); break;  case 6: DeleteData(); break;  case 7: Change(); break;  default:  break;  }  } while (var != 0);  }  default:  break;  }  } while (filevar != 0);  } |
| Данные из консоли |
| Файл Output.txt |
| Подобный файл будет создан и при выборе имени пользователем |