АКАДЕМИЯ УПРАВЛЕНИЯ ПРИ ПРЕЗИДЕНТЕ РЕСПУБЛИКИ БЕЛАРУСЬ

ИНСТИТУТ УПРАВЛЕНЧЕСКИХ КАДРОВ

Специальность: управление информационными ресурсами

**Алгоритмизация и программирование**

**Отчет**

**по лабораторной работе № 15**

|  |  |
| --- | --- |
| Выполнила:  Студентка  Группа: УИР-2  Курс: 1 | Кравчук Ксения Алексеевна |
| Приняла:  преподаватель | Юрча Ирина Александровна |

Минск 2022

**Цель работы:** изучить структурные шаблоны, структурные переменные, массивы структур, вложенные структуры и указатели на структуры, объединения, функцию typedef, научиться инициализировать структуры, передавать информацию о структурах функциям.

**Ход работы:**

**Ответы на контрольные вопросы:**

1. Что такое структурный шаблон, и как его определять?

Для объявления структуры используется структурный шаблон, который является основной схемой, описывающей содержание структуры. Общий формат объявления:

struct имя\_ типа\_ структуры {

тип имя\_элемента 1;

тип имя\_элемента 2;

тип имя элемента 3;

…

тип имя\_элемента М; }

2. Что такое имя структуры и как оно используется?

Необязательный «тег» (имя типа структуры) – слово stud, являющееся сокращенной меткой, которую можно использовать для ссылки на эту структуру. Имя структуры– это ее спецификатор типа.

3. Как определить структурную переменную?

Ключевое слово struct определяет, что все, что стоит за ним, является структурой.

4. Как обратиться к элементу структуры?

Обращаться в языке С (С++) к элементам структуры можно двумя способами:

Через операцию – «точку(.)» – обозначающий операцию получения элемента структуры. Обычная точка выполняет операцию доступа к элементам. Через операцию – «стрелку (–>)»

5. Как обратиться к указателю на структуру?

Операцию - стрелку используем, когда обращаемся к указателю на структуру.

6. Как обратиться к элементу при помощи указателя?

Обязательно надо ставить знак взятия адреса (&) перед именем структуры, так как имя структуры, в отличии от имени массива, не является указателем. Можно обратиться к элементам структуры через указатель другим способом: 12 printf("%.1f", (\*pmy).bal); При таком обращении надо брать в скобки указатель (\*ptr), т.к. операция доступа имеет более высокий приоритет по сравнению с операцией разыменования (\*)

7. Как передать в функцию элемент структуры?

Структуру можно передавать в функцию целиком, можно передать указатель на структуру, можно передать элементы структуры по отдельности.

8. Как сообщить функции о структуре?

Структура является типом данных созданным пользователем, к ней можно применять такие же операции, как и к встроенным типам.

9. Как создать вложенную структуру?

Для размещения вложенной структуры в шаблон она просто описывается точно так же, как это делалось бы для переменной типа int.

10. Как обратиться к элементу вложенной структуры?

Для размещения вложенной структуры в шаблон она просто описывается точно так же, как это делалось бы для переменной типа int. Чтобы обратиться к элементу вложенной структуры, нужно дважды использовать операцию точка «•».

11. Как создавать и использовать массивы структур?

Процесс описания массива структур совершенно аналогичен описанию любого другого типа массива. Чтобы объявить массив структур, необходимо сначала определить структуру, а затем объявить массив элементов этого структурного типа: struct stud my[5]; Этот оператор объявляет my массивом, состоящим из 5 элементов. Каждый элемент массива представляет собой структуру типа stud. Таким образом, my[0] является первой stud-структурой, my[1] – второй stud-структурой и т.д. Имя myсамо по себе не является именем структуры; это имя массива, содержащего структуры. Работа с массивами структур была приведена в примере 2 стр. 9-10.

12. Как создать объединение?

Одновременно с установкой шаблона (если на данную структуру больше не будет ссылок) можно сразу после объявления структуры объявить одну или несколько переменных. Процесс определения структурного шаблона и процесс определения структурной переменной объединяется в один этап. Объединение шаблона и определений переменных является именно тем случаем, когда не нужно использовать имя типа структуры.

13. Как использовать typedef?

На самом деле здесь не создается новый тип данных, а определяется новое имя существующему типу. Объявления typedef можно использовать для создания более коротких и содержательных имен стандартным типам данных, определенных в языке, или для типов, которые объявили пользователи

**Задания:**

6. Склад. Наименование товара, цена, количество, процент торговой надбавки (5, 10, 15, 20, 35, 30), страна производитель. Выбор по наименованию, цене, стране производителю. Вывод всего списка товаров на складе с расчетом их общей суммы. Результаты вывести на экран.

#include <iostream>

using namespace std;

struct product {

string name;

float cost;

int quantity;

string country;

};

void outputProduct(product products[], const int numOfProducts)

{

string name;

cout << "\nВведите имя нужного товара:";

cin >> name;

int i;

for (i = 0; i < numOfProducts; i++)

{

if (products[i].name == name)

{

cout << "\nИмя: " << products[i].name;

cout << "\nЦена: " << products[i].cost;

cout << "\nКоличество: " << products[i].quantity;

cout << "\nСтрана-производитель: " << products[i].country;

return;

}

}

cout << "К сожалению, нет товара с таким именем";

}

void add(product products[], int& index, const int numOfProducts)

{

string name;

if (index < numOfProducts) {

cout << "\nВведите имя товара:\n";

cin >> name;

products[index].name = name;

cout << "\nВведите стоимость товара:";

cin >> products[index].cost;

cout << "\nВведите кол-во данного товара:";

cin >> products[index].quantity;

cout << "\nВведите страну-производителя товара:";

cin >> products[index].country;

index++;

}

else

cout << "\nПростите, но склад не вместит больше товаров";

}

float calcAll(const product products[], const int numOfProducts)

{

float sum = 0;

float adding;

for (int i = 0; i < numOfProducts; i++)

{

cout << "\nВведите надбавку:";

cin >> adding;

sum += products[i].cost \* products[i].quantity \* adding / 100;

}

return sum;

}

void outputTask()

{

cout << "\nВарианты операций:";

cout << "\n1 - добавить товар";

cout << "\n2 - вывести данные о товаре";

cout << "\n3 - вывести информацию о товарах";

cout << "\n0 - выйти из программы";

cout << "\nВведите Ваш выбор:";

}

int main()

{

setlocale(LC\_ALL, "RUS");

const int NUM\_OF\_PRODUCTS = 5;

product sclad[5] = {};

int currentIndex = 0;

int choice;

do {

outputTask();

cin >> choice;

switch (choice) {

case 1: add(sclad, currentIndex, NUM\_OF\_PRODUCTS);

break;

case 2: outputProduct(sclad, currentIndex);

break;

case 3: cout << "\nСумма продуктов:" << calcAll(sclad, currentIndex);

break;

}

} while (choice != 0);

}

