

Министерство науки и высшего образования Российской Федерации

**Федеральное государственное автономное образовательное учреждение высшего
образования**

«Национальный исследовательский университет ИТМО»

Факультет информационных технологий и программирования

Лабораторная работа № 7

Типы данных, определяемые пользователем

Вариант 8

Выполнил студент группы № М3111

Гонтарь Тимур Сергеевич

Подпись:



Санкт-Петербург
2022

Условие ЛР:

1. Использовать перечислимый тип для хранения названий дней недели (месяцев, названий организационно-правовых форм предприятий и т.п.). Вывести на консоль значение целого числа, соответствующее указанному в варианте задания элементу перечисления.
2. Разработать структуру, описывающую координаты и другие требуемые параметры простейшей геометрической фигуры (отрезок прямой, прямоугольник, квадрат, треугольник, окружность) с расчётом одного из её параметров (периметра, площади, длины окружности и т.п.). Исходные данные для расчёта указать в тексте программы. Результат расчёта вывести на консоль.
3. Разработать битовое поле, описывающее состояния элементов прибора. Ввести с консоли число в 16-ричной системе счисления, преобразовать его в битовое поле за счёт использования объединения. Вывести на консоль состояния элементов прибора, соответствующие состоянию битов во введённом числе.

Решение с комментариями:

Мой вариант №8

В первом пункте я инициализирую перечислимый тип данных `enum`, в нем присваиваю целочисленные значения каждому типу ламп. Затем я создаю переменную, которая хранит соответствующее значение для галогенной лампы. Вывожу это на консоль.

В 2 пункте я создаю структуру `треугольник`, которая хранит координаты его вершин и его периметр. Я в функции `main` передаю координаты вершин треугольника, а затем запускаю написанную мной функцию для подсчёта периметра. Результат вывожу на консоль.

В 3 пункте я создаю объединение, которое объединяет в памяти целое число и структуру, в котором 4 битовых поля по 1 биту. В итоге я вывожу 4 самых «правых» бита числа.

```
1  #include "stdio.h"
2  #include "math.h"
3  #include "locale.h"
4
5  enum Lamps {
6      nakalivanya = 1,
7      day_light = 2,
8      gologen = 3,
9      neon = 4
10 };
11
12 struct Triangle {
13     double x1;
14     double y1;
15     double x2;
16     double y2;
17     double x3;
18     double y3;
19     double perimetr;
20 };
21
22 union CardReader {
23     int cr;
24     struct {
25         unsigned state: 1;
26         unsigned SDactive: 1;
27         unsigned CLactive: 1;
28         unsigned MSactive: 1;
29     };
30 };
31
32 double perimetr1(struct Triangle tr) {
33     double len1 = sqrt( (tr.x2 - tr.x1) * (tr.x2 - tr.x1) + (tr.y2 - tr.y1) * (tr.y2 - tr.y1) );
34     double len2 = sqrt( (tr.x3 - tr.x2) * (tr.x3 - tr.x2) + (tr.y3 - tr.y2) * (tr.y3 - tr.y2) );
35     double len3 = sqrt( (tr.x3 - tr.x1) * (tr.x3 - tr.x1) + (tr.y3 - tr.y1) * (tr.y3 - tr.y1) );
```

```

35 double len3 = sqrt( pow( (tr.x3 - tr.x1), 2) + pow( (tr.y3 - tr.y1), 2));
36 double p = len1 + len2 + len3;
37 return p;
38
39 }
40
41 int main() {
42     setlocale( LC_ALL, "Russian");
43     enum Lamps gal = gologen;
44     printf( format: "%d\n", gal);
45
46     double points[6] = { 1.0, 2.0, 3.0, 4.0, 5.0, 6.0};
47
48     struct Triangle tr1;
49     tr1.x1 = points[0];
50     tr1.y1 = points[1];
51     tr1.x2 = points[2];
52     tr1.y2 = points[3];
53     tr1.x3 = points[4];
54     tr1.y3 = points[5];
55     tr1.perimetr = perimetr1( tr1);
56
57     double ans = tr1.perimetr;
58     printf( format: "%lf\n", ans);
59
60     int values;
61     printf( format: "Введите число в 16-ричной системе счисления: ");
62     scanf( format: "%x", &values);
63     union CardReader cr1 = { values};
64     printf( format: "State: %d\n", cr1.state);
65     printf( format: "SD card active: %d\n", cr1.SDActive);
66     printf( format: "Compact flash active: %d\n", cr1.CLActive);
67     printf( format: "Memory stick active: %d\n", cr1.MSActive);
68
69     return 0;

```

Вывод: В ходе данной лабораторной работы я использовал перечислимый тип данных enum, работал со структурами, объединением а также с битовыми полями.