# Лабораторная работа № 11. Структуры и объединения

## Задание 1

1. Создать некоторую структуру с указателем на некоторую функцию в качестве поля. Вызвать эту функцию через имя переменной этой структуры и поле указателя на функцию.
2. –

|  |  |  |
| --- | --- | --- |
| Идентификатор | Семантика | Тип |
| s | Сумма | int |
| m, n | Слагаемые | int |
| function | Указатель на функцию | int |
| obj | Экземпляр структуры | somestruct |
| somestruct | Структура | struct |

#include <stdio.h>  
  
int somefunction(int m, int n) {  
 int s = m + n;  
 return s;  
}  
  
int main() {  
 struct somestruct {  
 int (\*function)(int m, int n);  
 } obj;  
 obj.function = somefunction;  
 int s = obj.function(2, 3);  
 printf("%d", s);  
 return 0;  
}

1. 

## Задание 2

1. Создать структуру дял вектора в 3-х мерном пространстве. Реализовать и использховать в своей программе следующие операции над векторами:

* скалярное умножение векторов;
* векторное произведение;
* модуль вектора;
* распечатка вектора в консоли.

В структуре вектора указать имя вектора в качестве отдельного поля этой структуры.

1. –
2. А

|  |  |  |
| --- | --- | --- |
| Идентификатор | Семантика | Тип |
| x, y, z | Переменные для подсчёта векторного произведения | int |
| vector | Структура | struct |
| obj1, obj2 | Экземпляры структур | (struct)vector |

#include <stdio.h>

#include <math.h>

int main() {

struct vector {

int x, y, z;

char name;

};

struct vector obj1 = {2, 3, 5, {'a'}};

struct vector obj2 = {5, 6, 1, {'b'}};

printf("Вектор %c = {%d; %d; %d}.", obj1.name, obj1.x, obj1.y, obj1.z);

printf("\nВектор %c = {%d; %d; %d}.", obj2.name, obj2.x, obj2.y, obj2.z);

printf("\n\nСкалярное произведение векторов = %d.", obj1.x\*obj2.x + obj1.y\*obj2.y + obj1.z\*obj2.z);

int x = obj1.y\*obj2.z - obj1.z\*obj2.y;

int y = -(obj1.x\*obj2.z - obj1.z\*obj2.x);

int z = obj1.x\*obj2.y - obj1.y\*obj2.x;

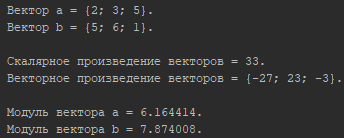
printf("\nВекторное произведение векторов = {%d; %d; %d}.", x, y, z);

printf("\n\nМодуль вектора %c = %lf.", obj1.name, sqrt(obj1.x\*obj1.x + obj1.y\*obj1.y + obj1.z\*obj1.z));

printf("\nМодуль вектора %c = %lf.", obj2.name, sqrt(obj2.x\*obj2.x + obj2.y\*obj2.y + obj2.z\*obj2.z));

return 0;

}

1. 

## Задание 3

1. Вычислить, используя структуру комплексного числа, комплексную экспоненту exp(z) некоторого z ∈ C: 
2. 
3. K

|  |  |  |
| --- | --- | --- |
| Идентификатор | Семантика | Тип |
| mycomplex | Структура | struct |
| i | Параметр цикла | int |
| n | Вводимое число | int |
| struct mycomplex z | Экземпляр структуры | struct |
| a, b | Действительные и мнимые части для комплексных чисел | double |

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

int factorial(int n) {

if (n == 0 || n == 1) return 1;

return n \* factorial(n - 1);

}

int main() {

struct mycomplex

{

double a;

double b;

} z;

int n, i;

double a, b;

srand(time(NULL));

z.a = rand() % 10;

z.b = rand() % 10;

printf("Введите n: ");

scanf("%d", &n);

printf("\nz = %lf + %lf \* i\n", z.a, z.b);

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

{

a += pow(z.a, n)/factorial(n);

b += pow(z.b, n)/factorial(n);

}

printf("exp(z): %lf + %lf \* i", a, b);

return 0;

}

1. 