**Работа № 2.  
Классы. Программирование линейных алгоритмов с использованием функций инициализации set() и вывода результатов print()**

**Теоретические сведения**

**Задание к работе**

**Варианты заданий**

**Цель работы –** изучить основные способы работы с пользовательским типом данных «класс», его объектами, методами и способы доступа к ним.

## Теоретические сведения

Основное отличие С++ от С состоит в том, что в С++ имеются классы. С точки зрения языка С классы в С++ - это структуры, в которых вместе с данными определяются функции. Это и есть инкапсуляция в терминах ООП.

Класс (class) - это тип, определяемый пользователем, включающий в себя данные и функции, называемые методами или функциями-членами класса.

Данные класса - это то, что класс знает.

Функции-члены (методы) класса - это то, что класс делает.

Таким образом, определение типа задаваемого пользователем (class) содержит спецификацию данных, требующихся для представления объекта этого типа, и набор операций (функций) для работы с подобными объектами.

Объявление класса

Приведем пример объявления класса

class my\_Fun{// компоненты-данные double x,y;  // компоненты-функции public:// функция инициализации void set(char \*c,double X){x=X;y=sin(x);    }// функция вывода результатов void print(void){cout << point<<y << endl;}};

Обычно описания классов включают в заголовочные файлы (\*.H), а реализацию функций-членов классов - в файлы \*.CPP.

Для каждого объекта класса устанавливается область видимости либо явно – указанием уровня доступа одним из ключевых слов public, private, protected с двоеточием, либо неявно – по умолчанию. Указание области видимости относится ко всем последующим объектам класса, пока не встретится указание другой области видимости. Область видимости public разрешает доступ к объектам класса из любой части программы, в которой известен этот объект (общедоступный). Область видимости private разрешает доступ к объектам класса только из методов этого класса. Объекты с такой областью видимости называют частными. Область видимости protected определяется для защищенных объектов, она имеет смысл только в иерархической системе классов и разрешает доступ к объектам этой области из методов производных классов. В теле класса ключевое слово области видимости может использоваться неоднократно. Область видимости для объектов типа «класс» по умолчанию private.

Способы объявления и инициализации объектов и доступ к методам класса:

1. Прямой вызов

my\_Fun Fun1; //объявление объекта1,но не инициализация  
Fun1.set("Function1 = ",1.0); // инициализация данных  
Fun1.print();                 // прямой вызов   
cout << "Input enter1..." << endl<<endl;

2. Косвенный вызов

my\_Fun \*p1 = &Fun1;  // воспользовались объектом 1  
                     // новая инициализация  
p1->set("Function1 = ",1.0); // косвенный вызов     
p1->print();                  // косвенный вызов     
cout << "Input enter1..." << endl<<endl;

3. Динамическое выделение памяти

my\_Fun \*p1 = new my\_Fun;  
p1->set("Function1 = ",1.0);  // косвенный вызов     
p1->print();                  // косвенный вызов     
cout << "Input enter1..." << endl<<endl;

// удаляется динамически выделенный объект  
delete p1;

## Задание к работе

Пользовательский класс должен содержать необходимые элементы-данные, метод установки их начальных значений:

Void set(double X, …);

метод печати:

Void print(void);

метод, решающий поставленную задачу:

Void Run(void);

Код методов – вне пространства определения класса. Программа должна включать в себя статический и динамический способы создания объектов, и для каждого объекта использовать прямую и косвенную адресацию при вызове методов класса.

## Варианты заданий

1. lab_clip_image035  
     
   При x=0.1722, y=6.33, z=3.25lab_clip_image037 **lab_clip_image039=-205.305.**