

## Лабораторная работа №1.

### «Простые объекты»

#### 1 Теоретическая часть.

В С++ так же, как и в других языках программирования, класс – создаваемый программистом структурный тип данных, который используется для описания множества объектов предметной области, имеющих общие свойства и поведение.

Класс объявляется следующим образом:

```
class <Имя класса>
{
private: <Внутренние (недоступные) компоненты класса>
protected: <Защищенные компоненты класса>
public: <Общие (доступные) компоненты класса>
};
```

Описание предусматривает три секции. Компоненты класса, объявленные в секции `private`, называются *внутренними*. Они доступны только компонентным функциям того же класса и функциям, объявленным *дружественными* описываемому классу.

Компоненты класса, объявленные в секции `protected`, называются *защищенными*. Они доступны компонентным функциям не только данного класса, но и его потомков.

При отсутствии наследования – интерпретируются как внутренние.

Компоненты класса, объявленные в секции `public`, называются *общими*. Они доступны за пределами класса в любом месте программы. Именно в этой секции осуществляется объявление полей и методов *интерфейсной части* класса.

Если при описании класса тип доступа к компонентам не указан, то по умолчанию принимается тип `private`.

В качестве компонентов в описании класса фигурируют *поля*, применяемые для хранения параметров объектов, и *функции*, описывающие правила взаимодействия с этими полями. В соответствии со стандартной терминологией ООП функции – компоненты класса или *компонентные функции* можно называть *методами*.

Компонентные функции или методы могут быть описаны как внутри, так и вне определения класса. В последнем случае определение класса должно содержать

прототипы этих функций, а заголовок описываемой функции должен включать *квалификатор видимости*, который состоит из имени класса и знака «::». Таким образом, компилятору сообщается, что определяемой функции доступны внутренние поля класса:

```
<Тип результата> <Имя класса> :: <Имя функции>(<Список параметров>)  
{  
    <Тело компонентной функции>  
}
```

#### **Пример 1.1** Описание класса.

##### **А.** Описание компонентных функций внутри класса.

```
#include <stdio.h>  
class First  
{  
public:  
    char c;  
    int x,y;  
    /* компонентные функции, определенные внутри класса */  
    void print(void)  
    {  
        printf ("%c %d %d ",c,x,y);  
    }  
    void set(char ach,int ax,int ay)  
    {  
        c=ach; x=ax; y=ay;  
    }  
};
```

##### **Б.** Описание компонентных функций вне класса.

```
#include <stdio.h>  
class First  
{  
public: char c;  
        int x,y;  
    void print(void);
```

```

void set(char ach,int ax,int ay);
};
/* компонентные функции, описанные вне класса */
void First::print(void)
{   printf ("%c %d %d ",c,x,y); }
void First::set (char ach,int ax,int ay)
{   c=ach; x=ax; y=ay; }

```

*Конструктор* – метод класса, который *автоматически вызывается при выделении памяти под объект*.

По правилам C++ *конструктор* имеет то же имя, что и класс, не наследуется в производных классах, может иметь аргументы, но не возвращает значения, может быть параметрически перегружен. Конструктор определяет операции, которые необходимо выполнить при создании объекта. Традиционно такими операциями являются инициализация полей класса и выделение памяти под динамические поля, если такие в классе объявлены. Явный вызов конструктора не возможен, что в некоторых случаях усложняет создание инициализированных объектов.

При освобождении объектом памяти автоматически вызывается другой специальный метод класса – *деструктор*. Имя деструктора по аналогии с именем конструктора, совпадает с именем класса, но перед ним стоит символ «~» («тильда»). Деструктор определяет операции, которые необходимо выполнить при уничтожении объекта. Обычно он используется для освобождения памяти, выделенной под динамические поля объекта данного класса конструктором, и при необходимости может быть объявлен виртуальным.

Деструктор не возвращает значения, не имеет параметров и не наследуется производными классами. Класс может иметь только один деструктор или не иметь ни одного. В отличие от конструктора деструктор может вызываться явно.

**Пример 1.2** Создание, инициализация и уничтожение объекта при наличии в классе конструктора и деструктора.

```

#include <locale.h>
#include <iostream>
using namespace std;
class Num
{
    int n;

```

```

public:
    Num(int an){ cout<<"Конструктор"<<endl; n=an; }
    ~Num() { cout<<"Деструктор"<<endl; }
};
void main(int argc, char* argv[])
{
    setlocale(0,"russian");
    Num N(56);
    system("pause");
}

```

Результат выполнения программы:

Конструктор

Для продолжения нажмите любую клавишу . . .

Деструктор

## 2 Практическая часть.

Необходимо описать класс, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанной структурой.

Задание по номеру в списке журнале.

### 2.1 Пример выполненного задания.

Задание: Объект – целое число. Параметр: значение. Методы – конструктор и метод, выводящий на экран значение числа. Вывести на экран значения всех чисел массива и посчитать их сумму.

```

#include "stdafx.h"
#include <conio.h>
//Объявление класса
class Num
{
private:
    int n; // атрибут "значение"
public:
    //конструктор
    Num (int an){
        n = an;
    }
    // метод, выводящий значение на экран
    void Print(){
        printf("%d\n",n);
    }
}

```

```

}
int getNum() {
    return n;
}
};
void main(int argc, char* argv[])
{
    int Sum = 0;
    Num N[5] = {1,5,6,3,9}; // инициализация массива объектов
    for(int i=0;i<5;i++)
    {
        N[i].Print(); //вызов метода вывода значения на экран
        Sum += N[i].getNum(); // суммирование
    }
    printf("Sum = %d", Sum);
    _getch();
}

```

### **3 Защита.**

**Для защиты необходимо предоставить работающую программу, диаграмму классов, знать теорию.**

### ***Вариант 1.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект - вещественное число, записанное с точкой. Параметры - его значение и запись. Методы: конструктор и метод, определяющий количество цифр в дробной и целой части числа. Определить сумму введенных чисел и отношение количества цифр в целой и дробной частях.

### ***Вариант 2.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект – прямоугольник. Параметры – длины его сторон. Методы: конструктор и метод, определяющий площадь прямоугольника. Определить количество прямоугольников с площадью больше 50.

### ***Вариант 3.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект – боевой хомяк. Параметры – количество побед и поражений. Методы: конструктор и метод, определяющий эффективность хомяка. Определить хомяка с наибольшей эффективностью.

### ***Вариант 4.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект – целое число. Параметры - его значение и запись. Методы: конструктор и метод, определяющий четное число или нет. Определить сумму нечетных чисел.

### ***Вариант 5.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект – объект. Параметры – целое число и целое число. Методы: конструктор и метод, определяющий наибольшее число. Определить среднее значение наибольших чисел.

### ***Вариант 6.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект - некоторый товар в магазине. Параметры: наименование, количество и закупочная цена. Методы: конструктор и метод, определяющий стоимость товара исходя из заданного процента прибыли. Определить суммарную стоимость всех товаров в магазине.

### ***Вариант 7.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект - некоторый товар в магазине. Параметры: наименование, количество и стоимость. Методы: конструктор и метод, определяющий суммарную стоимость товара. Определить суммарную стоимость всех товаров в магазине.

### ***Вариант 8.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект - скаковая лошадь. Параметры: кличка и массив рекордов, содержащий 5 лучших результатов, показанных лошадью на скачках. Методы: конструктор и метод, определяющий среднее время, показанное лошадью. Определить среднее время по всей конюшне.

### ***Вариант 9.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект - ангар. Параметры: ширина и длина. Методы: конструктор и метод, определяющий площадь помещения. Определить площадь склада, состоящего из нескольких ангаров.

### ***Вариант 10.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект - ангар. Параметры: площадь и процент занятой площади. Методы: конструктор и метод, определяющий свободную площадь помещения. Определить свободную площадь склада, состоящего из нескольких ангаров.

### ***Вариант 11.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект - квартира. Параметры: общая площадь и стоимость одного метра. Методы: конструктор и метод, определяющий стоимость квартиры. Определить все квартиры, стоимость которых не превышает заданной суммы.

### ***Вариант 12.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект - квартира. Параметры: стоимость и количество комнат. Методы: конструктор и метод, определяющий среднюю стоимость одной комнаты. Определить все квартиры, стоимость одной комнаты в которых не превышает заданной суммы.



### ***Вариант 13.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект – боевой хомяк. Параметры – количество побед и поражений. Методы: конструктор и метод, определяющий эффективность хомяка. Определить эффективность боевой группы.

### ***Вариант 14.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект - вещественное число, записанное с точкой. Параметры - значение числа и его запись. Методы: конструктор и метод, определяющий количество цифр в целой части числа. Определить среднее значение введенных чисел и среднее количество цифр в целой части.

### ***Вариант 15.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект – прямоугольник. Параметры – длины его сторон. Методы: конструктор и метод, определяющий периметр прямоугольника. Определить прямоугольник с наименьшим периметром.

### ***Вариант 16.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект - некоторый товар в магазине. Параметры: наименование, количество, процент затрат на реализацию и закупочная цена. Методы: конструктор и метод, определяющий себестоимость товара для магазина. Определить суммарную себестоимость всех товаров в магазине.

### ***Вариант 17.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект – целое число. Параметры - его значение и запись. Методы: конструктор и метод, определяющий четное число или нет. Определить количество четных чисел.

### ***Вариант 18.***

Описать объект, включающий заданные поля и методы. Написать программу, которая создает массив объектов и содержит процедуры, работающие с указанными структурами.

Объект – прямоугольник. Параметры – длины его сторон. Методы: конструктор и метод, определяющий площадь прямоугольника. Определить прямоугольник с наибольшей площадью.