1. Ctrl+F5执行不调试，F5调试
2. C++标准输入输出流，数据从键盘流向程序为输入，从程序流向屏幕或磁盘为输出。输出流cout，可以用多个插入运算符<<连接输出流。输入流cout，可以用多个提取运算符>>连接输入流。
3. 按位异或^，按位取反~

位运算不使用中间变量交换变量：

int a = 10;int b = 12;

a = a ^ b;

b = a ^ b;

a = a ^ b;

1. 函数重载，函数名相同，而参数表（参数个数或类型）不同，通过参数表唯一标识并且来区分函数的一种特殊函数。
2. 二维数组数组名代表数组的首地址，也就是第0号元素的地址。指针接收二维数组的地址，是第一行的首地址，而不是数组名。如p=array[0]，而不是p=array。
3. 使用运算符new在堆内存区中进行内存的动态分配。

double \* d; d=new double; \*d=30.5;

double \* d; d=new double(30.5);

int n;

int \* p;

//输入n值

p=new int[n];

1. C语言中返回值0代表正常，非0代表异常。
2. 运算符delete用来释放（动态回收）new请求到的内存空间。

Delete d; d= NULL;

释放数组：delete [n]p; p=NULL;

1. 结构体，是一种类型，不分配内存

定义变量方法1：

struct 结构体名{

类型 成员名;

类型 成员名;

…

};

struct 结构体名 变量名;

定义变量方法2：

struct 结构体名{

类型 成员名;

类型 成员名;

…

}变量名;

定义变量方法3：

struct {

类型 成员名;

类型 成员名;

…

}变量名;

1. 共用体，即联合体，使用用一段空间，是一种类型，不分配内存

定义变量时会选取最长成员占用的字节数开辟空间

定义变量方法1：

union共用体名{

类型 成员名;

类型 成员名;

…

};

union共用体名 变量名;

定义变量方法2：

union共用体名{

类型 成员名;

类型 成员名;

…

}变量名;

定义变量方法3：

union {

类型 成员名;

类型 成员名;

…

}变量名;

1. 面向过程程序设计：程序=算法+数据结构

面向对象程序设计：对象=算法+数据结构，程序=对象+对象+对象…

1. C++的基础-类：是一个抽象的过程，对具体的对象概括，提取共用的东西

类描述了一组有相同特性（数据元素）和相同行为（函数）的对象。*类生成的实例就是对象。类和类型差不多，对象是用类定义的一个变量，它占用内存，是程序的实体。*

1. 类声明，放在头文件中，类是一种数据类型，不分配内存，不能对数据成员初始化。

class <类名>

{//下面定义访问权限，在类中private, public, protected可以重复使用

private: //类内成员或友元访问，不能外部（用户代码）访问

[私有型数据和函数]

public://类外（用户代码）可以访问

[公有型数据和函数]

protected: //类中使用，子类（派生类）访问，不能外部（用户代码）访问

[保护型数据和函数]

}

vc2015右键点击工程中“源文件”->“添加”->“类…”，创建一个“C++类”。类名大写字母C开头代表类。自动生成构造函数和析构函数两个函数。

1. 类的实现，各成员函数的实现

<函数类型><类名>::<函数名>(<形式参数>)

{

函数体;

}

1. 大部分成员变量设置为私有private，大部分成员函数设置为公有的public
2. 对象定义
3. 定义对象方法1：声明之后定义，推荐这种方法

class A{

…

};

class A 对象名;

~~定义对象方法2：声明时定义~~

~~class A {~~

~~…~~

~~}变量名;~~

~~定义对象方法3：一次性定义~~

~~class {~~

~~…~~

~~}变量名;~~

1. 啊