1. C++中不能重载的六个运算符号：.，：：，？，sizeof，typeid，\*
2. 构造函数是否可以是虚函数？析构函数呢？

答：构造函数不可以，虚函数是借助于虚函数表，而且是通过虚函数指针找到虚函数表，然而虚函数指针存在于对象之中，要有个对象才行，而调用构造函数的时候对象还没有建立，没有建立怎么会有虚函数指针呢？所以不可以。

析构函数可以当一个父类的析构函数被声明为虚函数的时候，我们delete一个指向子类对象的父类指针的时候，实际调用的是子类的析构函数，但是子类析构函数会自动引发父类的析构函数的调用，从而使子类特有的部分和从父类继承的部分能够释放，我们一般把父类的析构函数声明成虚函数，尽管这个析构函数什么都不做。（最好是把基类的析构函数定义为虚函数，否则当用基类去创建一个指向派生类的指针，当其释放的时候基类中的东西可能不会被释放，造成内存泄漏）

3.虚函数是否可以是内联函数

答：不可以，虚函数是通过虚函数表，这是在运行的时候确定的，然而内联函数的替换是在编译阶段，编译的时候不知道调用哪一个虚函数所以不可以。（编译时静态的，运行是动态的）

5静态成员函数是否可以是虚函数？

答：不可以，静态成员函数不属于哪个类，而虚函数通过虚函数指针找到虚函数表，虚函数依赖于对象指针，而静态函数不属于某各类，所以不可以。

6虚函数表和虚函数指针在运行，编译时建立？

答：虚函数表是在编译时建立，在编译的时候把类中的虚函数地址放到虚函数表中，构成一个指针数组。

虚函数指针是在运行时建立的，他也是类的隐藏的成员，他在调用构造函数的时候执行，构造函数执行完以后虚函数指针初始化好（构造函数在运行时调用的）

1. const与defeine的区别？

答：const有数据类型而define没有编译器对const进行类型安全检查，而对define只是简单的字符替换，没有安全检查，而且会引起错误。

1. 什么是虚函数？纯虚函数？什么是抽象类？纯抽象类？纯抽象类的作用？

答：虚函数：被virtual修饰的成员函数。（构造函数，静态成员函数和全局函数不能被声明为虚函数）

纯虚函数：virtual 函数返回值 函数名（）= 0

抽象类：有纯虚函数的类是抽象类。（不能创建抽象类的对象，当做一个接口类使用）

纯抽象类：除了构造函数与静态成员函数意外都是纯虚函数的类。

1. new、delete、malloc、free关系？

答：delete会调用对象的析构函数,和new对应free只会释放内存，new调用构造函数。malloc与free是C++/C语言的标准库函数，new/delete是C++的运算符。它们都可用于申请动态内存和释放内存。对于非内部数据类型的对象而言，光用maloc/free无法满足动态对象的要求。对象在创建的同时要自动执行构造函数，对象在消亡之前要自动执行析构函数。由于malloc/free是库函数而不是运算符，不在编译器控制权限之内，不能够把执行构造函数和析构函数的任务强加于malloc/free。因此C++语言需要一个能完成动态内存分配和初始化工作的运算符new，以及一个能完成清理与释放内存工作的运算符delete。注意new/delete不是库函数。

1. delete与 delete []区别？

答：当做用于一个数组时，delete只调用一次析构函数，而delete[]释放每一个的内存，调用每一个析构函数。Delete与new配合，delete[]与new[]配合。

1. 函数指针？

答：函数指针的生命更只需呀把指针替换函数名就可以。

void \* ( \* (\*fp1)(int))[10];

float (\*(\* fp2)(int,int,int))(int);

int (\* ( \* fp3)())[10]();

分别表示什么意思？

1.void \* ( \* (\*fp1)(int))[10]; fp1是一个指针，指向一个函数，这个函数的参数为int型，函数的返回值是一个指针，这个指针指向一个数组，这个数组有10个元素，每个元素是一个void\*型指针。

2.float (\*(\* fp2)(int,int,int))(int); fp2是一个指针，指向一个函数，这个函数的参数为3个int型，函数的返回值是一个指针，这个指针指向一个函数，这个函数的参数为int型，函数的返回值是float型。

1. int (\* ( \* fp3)())[10](); fp3是一个指针，指向一个函数，这个函数的参数为空，函数的返回值是一个指针，这个指针指向一个数组，这个数组有10个元素，每个元素是一个指针，指向一个函数，这个函数的参数为空，函数的返回值是int型。
2. 返回值是引用类型的函数是左值。
3. 栈的空间一般为2M
4. 内存泄漏是用完了却没有释放，所以系统可用的少了，内存泄漏
5. 类如果是空的系统会有个char字符来区分每个类，所以空类大小为1
6. 继承会继承所有的东西，比如私有成员，只是不能访问而已
7. 递归函数不能定义为内联函数，内联函数适用于不存在witch/switch，1-5行语句
8. Void \*p是空指针可以操作任意类型的运算符，sizeof(p)是指针大小，4(32位)8(64位)
9. 内联函数编译生成可执行文件时插入代码，会增加代码大小，提高运行速度
10. 小端储存 ：从低到高 大端储存：从高到底 如：0x20150810

存储地址： 10 08 15 20 大端存储：20 15 08 10

输出从高到低：08 10 20 15 输出低->高：20 15 08 10

1. 内存分配：

Malloc函数：void \*malloc(unsigned int size)

分配一个size大小的空间并返回首地址，不会初始化，里面都是随机的数

Calloc函数 ：void \*calloc(unsigned int num,,int size)

分配一个size\*num大小的空间，自动初始化内存空间为0

Realloc函数 void \*realloc(void \*ptr,unsigned int size)

分配一个长度为size的内存地址，把手地址给ptr不初始化

New运算符：自动计算所分配空间大小，分配类型同时调用构造函数。不初始化，除了static，因为static默认为0；

21.子类析构时要调用父类的析构函数吗？

析构函数调用的次序是先派生类的析构后基类的析构，也就是说在基类的的析构调用的时候,派生类的信息已经全部销毁了。定义一个对象时先调用基类的构造函数、然后调用派生类的构造函数；析构的时候恰好相反：先调用派生类的析构函数、然后调用基类的析构函数。

1. extern ,在.h中有个int a,如果在.cpp中需要调用则需extern int a,编译的时候当前文件找不到就回去其他文件找
2. Int \*p[10] p包含10个指针的数组int (\*p)[10]p指向含有10个整数的数组
3. 函数参数是顶层const时候相同，底层时候不同
4. 内层定义了一个变量之后，会隐藏外层同名变量
5. Int (\*pf)(int a)这是一个函数指针，int \*pf(int a这是一个函数，返回值是int \*
6. 构造函数加=default时表示，需要系统默认的构造函数
7. 委托构造函数：让别的构造函数替自己完成构造Studnet():Student()
8. explicit生命的构造函数只能直接初始化（int a=5拷贝初始化,int a(5),直接初始化）入比如

Student a(b)对，Studnet a=b;不对，

1. Final virrtual override 三个区别 带有final的类表示禁止集成，带有final的函数表示禁止重写
2. Constexpr 函数的参数和返回值必须是字面值，但是当一个构造函数是constexpr类型的时候不可以有返回值，所以可以在函数后面加个=default
3. assert是一个预处理宏，当里面的条件成立不起作用，不成立输出错误信息
4. const对象只能调用const函数，非const对象可以调用const函数
5. C++中有成员是被继承的只是不能引用
6. Int arr[][私4]这种的不初始化是错误的，初始化的话就对
7. “a”占两个字节因为他是字符串后面有”\0”
8. Int a=5,b=4,c=3 if(a>b>c)是对的，但是a>b之后是1>c?
9. 纯虚函数相当于接口，不能直接实例化，需要派生类来实现函数定义(virtual void fun()=0)
10. 如果基类有纯虚函数，而其派生类中没有重写改函数，则其依然为纯虚函数，这个派生类依然是个抽象类。
11. 数组/数组名作为函数的参数会蜕化成指针 a(int \*p) a(int a[]) a(int a[10])
12. .h头文件中的ifndef/define/endif 的作用？
13. 父类的析构函数是虚函数有何用哪个。是为了释放父类是可以释放子类C++的多态肯定是使用父类的指针指向子类的对象，所以肯定是释放子类的对象，如果不使用虚函数的话， 父类的指针就只能够释放父类的对象。
14. 逻辑与 第一个条件为假或真都不会去判断第二个了
15. 左定值，右定向，const int \*p p指向内存的值不可以改变，但是p可以换个指向地址

Int const \*p,\*p始终只能指向一个地方

1. 关于结构体：C中的结构体不可以继承，因为没有面向对象的思想，但是在C++中可以。默认访问继承权限是public,类的则是private,结构体也可以和类一样有构造函数
2. Char str[10];cin>>str;输入”I Love China”时只存入了I，因为cin>>遇到空格结束
3. 不能声明为虚函数的有：静态成员函数，类外的普通函数，构造函数，友元函数
4. 初始化列表要注意，按类中声明变量的先后顺序来初始化，与初始化列表的顺序无关。
5. 引用是除指针外另一个可以产生多态效果的手段，基类的引用可以指向派生类B &c=A
6. Printf(“%d”,a,b);参数入栈从右往左，打印还是先打印左边的
7. 8.9e1.2 不对，e前比有数，e后必是整
8. 未定义的数组成员默认是0
9. Get(3)遇到空格不会结束，遇到回车结束，所以可以输入带有空格的字符串
10. 抽象类不能建立对象，只能声明指针或者引用
11. #defin是预处理命令，预、编译程序完成的实际上是对源程序中部分内容的替代，经过替代生成一个没有宏定义，没有编译指令的文件，在编译预处理时候进行简单的替换不做正确性检测，不管是否正确都带入
12. 字符串初始化时候赋值的字符串被认为是常量不可以修改
13. #include包括""和<>这两种情况，""实在用户工作目录下寻找（用户的工作目录是通过编译器指定的），<>是找系统标准库函数，通过系统环境变量指定系统库目录；