1. 虚函数的存在有两个关键点，一个是在基类中被关键词virtual说明，另一个是派生类中重新定义，相同函数名。
2. 重载函数的参数个数或参数类型必须不同。而虚函数要求函数名，返回类型，参数个数，参数的类型与顺序与基类中的虚函数原型完全相同。
3. 如果一个类至少有一个纯虚函数，那么就称这个类为抽象类。
4. C++中绝大部分的运算符允许重载,不能重载的运算符只有少数几个。
5. C+语言中只能对已有的C++运算符进行重载,不允许用户自己定义新的运算符。(3）运算符重载是针对新类型数据的实际需要,对原有运算符进行适当的改造完成的。一般来讲,重载的功能应当与原有的功能相类似(如用“+”实现加法,用“-”实现减法)。
6. 重载不能改变运算符的操作对象(即操作数)的个数。(5)重载不能改变运算符原有的优先级。
7. 重载不能改变运算符原有的结合特性。
8. 运算符重载函数的参数至少应有一个是类对象(或类对象的引用)。
9. 运算符重载函数可以是普通函数﹐也可以是类的成员函数,还可以是类的友元函数。
10. 一般而言,用于类对象的运算符必须重载,但是赋值运算符“=”例外,不必用户进行重载.
11. 虚函数就是在基类中被关键字virtual说明,并在派生类中重新定义的函数。虚函数的作用是允许在派生类中重新定义与基类同名的函数,并且可以通过基类指针或引用来访问基类和派生类中的同名函数。
12. 在一个派生类中重新定义基类的虚函数是函数重载的另一种形式,但它不同于一般的函数重载。当普通的函数重载时,其函数的参数或参数类型必须有所不同,函数的返回类型也可以不同。但是，当重载一个虚函数时,也就是说在派生类中重新定义虚函数时,要求函数名、返回类型、参数个数、参数的类型和顺序与基类中的虚函数原型完全相同。如果仅仅返回类型不同,其余均相同,系统会给出错误信息;若仅仅函数名相同,而参数的个数、类型或顺序不同,系统将它作为普通的函数重载,这时虚函数的特性将丢失。