1. Чисто виртуальный метод содержит признак = 0 вместо тела. Также чисто виртуальный метод должен переопределяться в производном классе (возможно, снова как чисто виртуальный). Чисто виртуальный метод - это виртуальная функция, которая должна быть реализована производным классом, если производный класс не является абстрактным.
2. Это класс, содержащий хотя бы один чисто виртуальный метод
3. Абстрактные классы нужны для представления общих свойств, которые предполагается конкретизировать в производных классах
4. Полиморфные функции работают с объектом любого типа в пределах одной иерархии
5. Полиморфизм – это возможность объектов с одной и той же спецификацией иметь разную реализацию. А принцип подстановки реализует одинаковую спецификацию. Выходит, что принцип подстановки характеризует спецификацию классов, а полиморфизм – их поведение
6. class Figure

{

public:

    virtual double getSquare() = 0;

    virtual double getPerimeter() = 0;

    virtual void showFigureType() = 0;

};

1. struct Person  
   {  
   public:  
           ~Person() {}  
   private:  
           string name;  
           //...  
   };  
     
   struct Student : Person  
   {  
   public:  
           Student()  
           {  
                   someData = new Data();  
           }                 
           ~Student()  
           {  
                   delete someData;  
           }  
           //...  
   private:  
           Data \*someData;  
   };
2. В некоторых программах невозможно знать наперёд, какая функция будет вызываться первой. В таком случае используется позднее связывание (или «динамическая привязка»). В C++ для выполнения позднего связывания используются указатели на функции