

C++ course

name: Victoria

e-mail: kewtree1408@gmail.com

C++ course

source: <https://bitbucket.org/Kewtree/coursecpp/src>

login: coursecpp

password: 12345

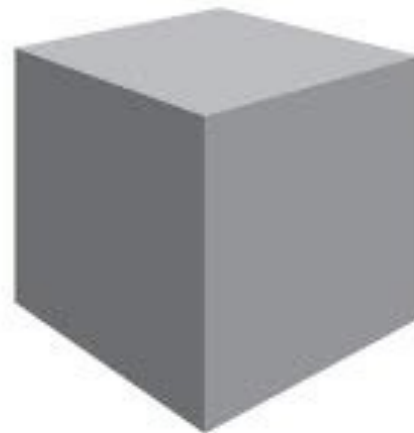
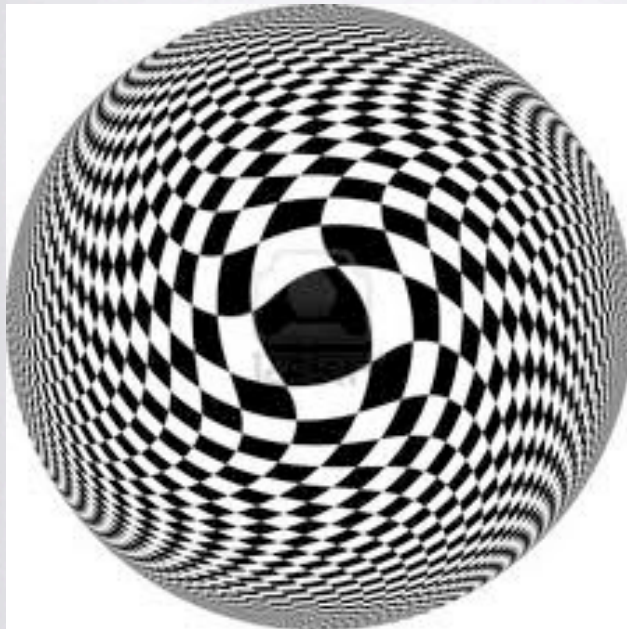
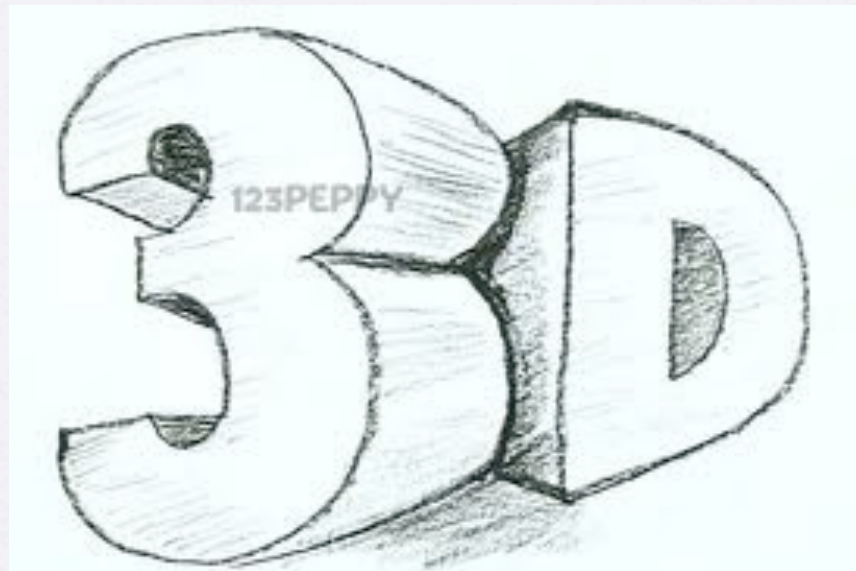
Plan 10

- Polymorphism
- Example: draw 3D
- Virtual functions, simple example
- Linking
- Abstract class
- Pure virtual function
- Example with abstraction class, polymorphism
- Virtual destructor

Polymorphism

- Один интерфейс -- множество реализаций.
- Возможность объектов с различной спецификацией иметь различную реализацию.

Example: draw 3D

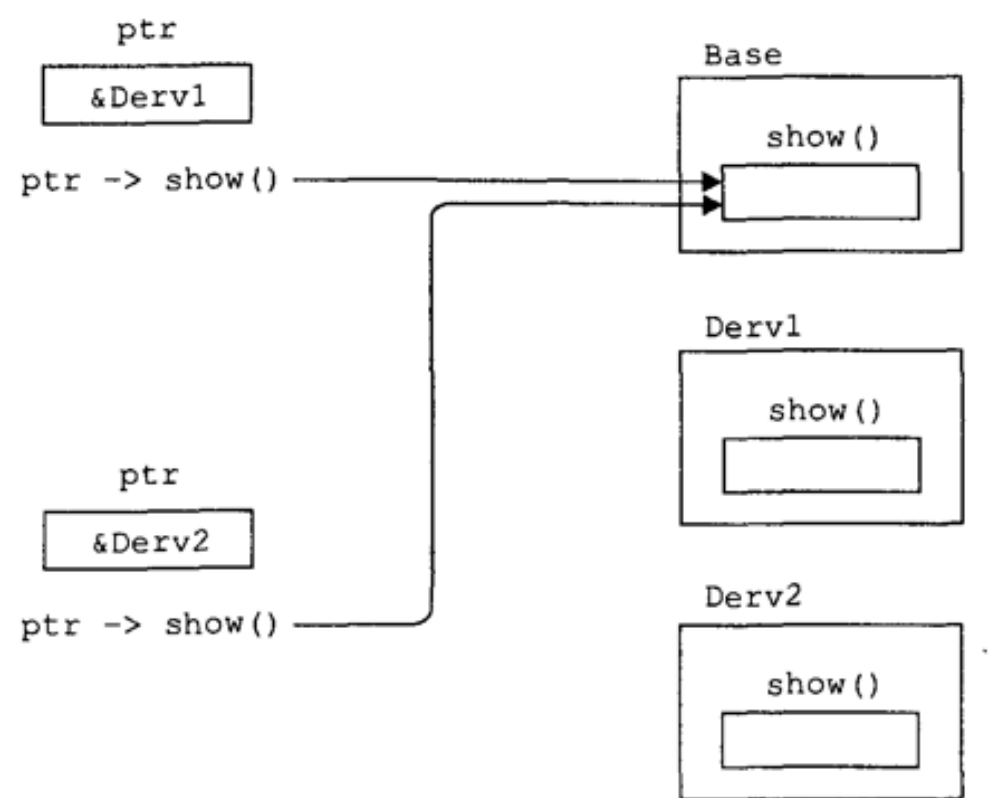


Virtual function

- Как добиться полиморфизма?
 - Использовать наследование
 - Использовать виртуальные функции
 - `virtual void hello() { cout << "Hello, world" << endl; }`

Virt-func: simple ex.

- simple_virtual.cpp



11.1. Доступ через указатель без использования виртуальных функций

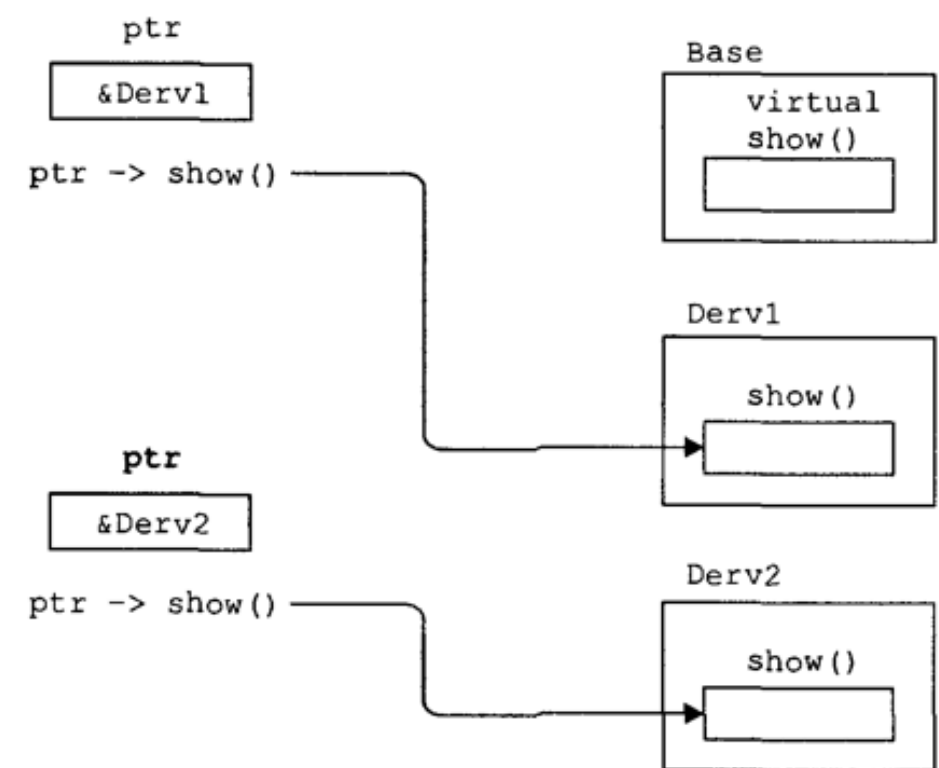


Рис. 11.2. Доступ через указатель к виртуальным функциям

Linking

- позднее связывание (динамическое)
- раннее связывание (статическое)

Abstract class

- Абстрактный класс -- базовый класс, объекты которого никогда не будут реализованы.
- Зачем такой класс нужен?
 - Быть родительским классом по отношению к производным
 - Быть звеном в иерархической структуре

Pure virtual function

- `virtual void show() const = 0;`
- объект нельзя создать
- указатель можно объявить

Example with abstract class+polymorphism

- `person.cpp`

Virtual destructor

- Нужно ли добавлять слово `virtual` в деструкторах производных классов?
- А в конструкторах?

Example with 2D

- draw with *

```
      *
    *
  **
 ***
****
*****

      *
    * *
  *   *
 *     *
*       *
 *     *
  *   *
    * *
      *

*****
*****
*****
*****
*****

*
**
***
****
*****
```