

Тема 2.9 Виртуальные классы и функции

Язык C++

Виртуальные функции

- *Виртуальный метод* – метод, который может быть переопределён в классах-наследниках так, что конкретная реализация метода для вызова будет определяться во время исполнения.
- Виртуальные методы описываются с помощью ключевого слова *virtual* в базовом классе
- Виртуальные методы не могут быть статическими
- Виртуальные методы позволяют создавать различные версии функции для базового и производных классов, однако должны иметь всегда одинаковое число и одинаковые типы параметров. В противном случае функция будет перегруженной, а не виртуальной.
- Когда виртуальная функция вызывается с указанием области видимости, например, `Point::Show()`, виртуальный механизм не применяется.

Виртуальные, чистые виртуальные функции и абстрактные классы

- Для отражения в реализации иерархии классов ситуации, при которой одна и та же функция базового класса должна различным образом выполняться в классах-потомках, служат **виртуальные** функции. Классы, включающие виртуальные функции, называются **полиморфными**. Виртуальными могут быть только нестатические компонентные функции классов
- Компонентная виртуальная функция (базового) класса, не имеющего в данном классе реализации, называется **чистой виртуальной** функцией, а класс, содержащий хотя бы одну чистую виртуальную функцию, — **абстрактным**

Понятие наследования в ООП и его реализация в Си++ (продолжение)

- Производному классу в языке Си++ доступны общедоступные (`public`) и защищенные (`protected`) компоненты базовых классов. С учетом вариантов спецификации доступа к компонентам базовых классов из производных, а также спецификации доступа в самих базовых классах, итоговая картина прав доступа имеет вид, показанный на следующем слайде
- При допустимом языком Си++ множественном наследовании возможно дублирование объектов непрямых базовых классов в составе общего предка. В тех случаях, когда такое дублирование нежелательно, базовый класс можно объявить **виртуальным**

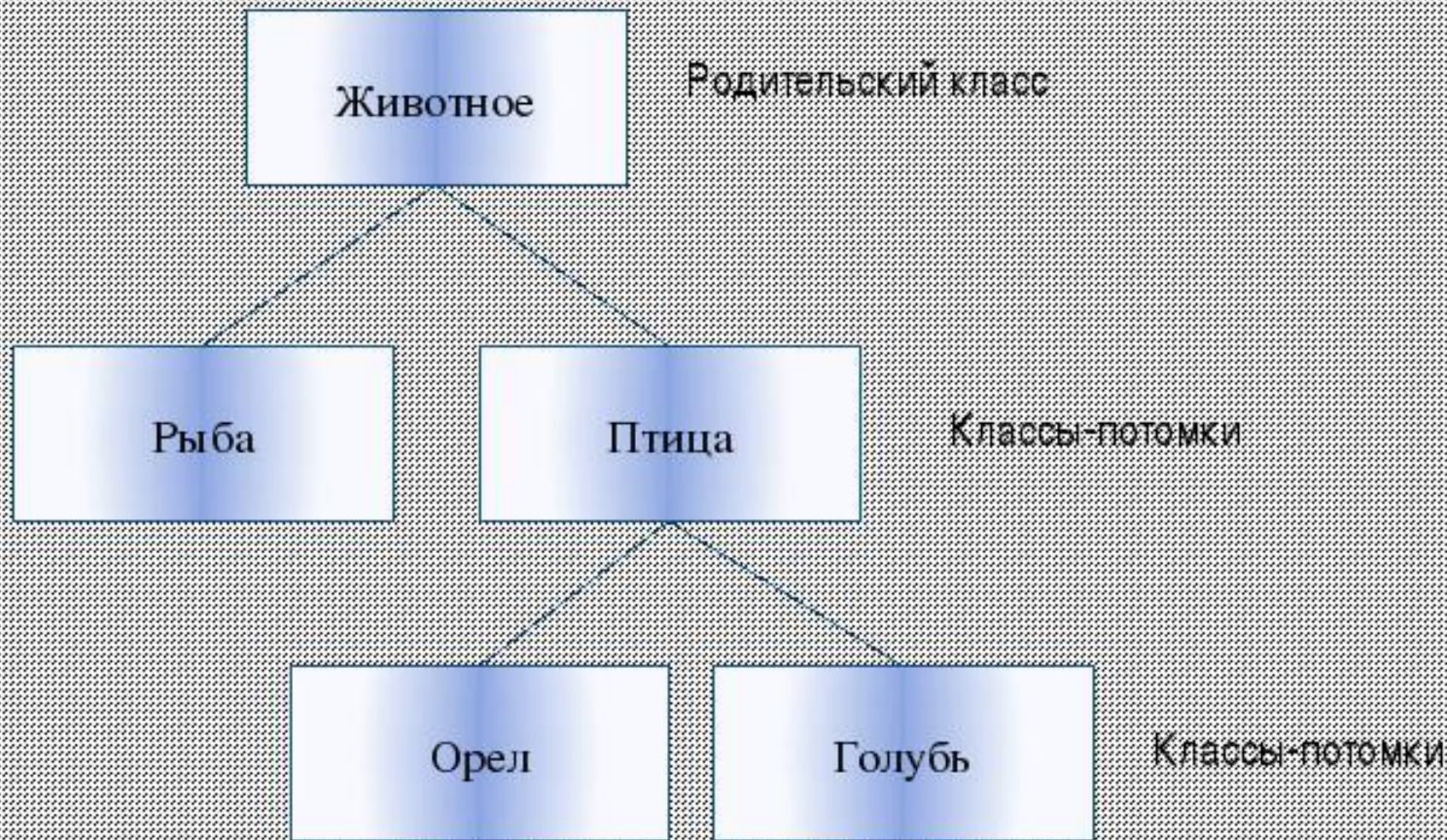
Ограничения виртуального наследования

- Классы-предки не могут одновременно перегружать одни и те же методы своего родителя
 - В нашем случае – нельзя переопределять метод Eat() одновременно и в C**Mammal**, и в C**WingedAnimal** – будет ошибка компиляции
 - В случае переопределения этого метода в одном из классов компилятор выдаст предупреждение

Понятие наследования в ООП и его реализация в Си++ (окончание)

Спецификатор в списке базовых классов	Спецификатор в теле базового класса	Доступ в производном классе
не указан	private	недоступны
не указан	protected	private
не указан	public	private
private	private	недоступны
private	protected	private
private	public	private
protected	private	недоступны
protected	protected	protected
protected	public	protected
public	private	недоступны
public	protected	protected
public	public	protected

Графическое изображение иерархий наследования



Конец