

Пример с човек, студент и преподавател.	
11	Статично и динамично свързване. Виртуални функции. Ключови думи - override, final. Виртуални таблици. Полиморфизъм. Абстрактни класове. Пример за фигури.
12	Многократно наследяване. Полиморфизъм.

Статично и динамично свързване

- Статично - изборът на ф-ята, която трябва да се изпълни става по време на компилация
 - определя се от типа на из-я/референцията, от която се извиква ф-ята
- Динамично - изборът на ф-ята, която трябва да се изпълни става по време на изпълнение на програмата

Полиморфизъм

def / име на ф-я, много имплементации

Видове:

- Compile-time - function and operator overloading
- Run-time - динамично свързване

Виртуални функции - чрез тях осъществяваме динамично свързване

- чрез ключовата дума virtual означаваме, че съответна ф-ция може да се презаписва от наследниците на класа (при наличие на override се прави и проверка на синтаксиса)
- override - означаваме, когато презаписваме виртуална ф-я
- final → за виртуални ф-ии - забранява да се презаписва повече функцията
 - за класове забранява наследяването на класа

Синтаксис:

```
class Base {
    virtual void f();
}
```

```
class Der : Base
{
```

```
    void f() override;
    ↳ Der::f() също е виртуална,
    защото виртуалността се
```

}

↳ Der::f() също е виртуален, защото виртуалността се наследява

По този начин можем да извикваме ф-ии от наследник през базов клас.

! При полиморфна йерархия деструкторът на Base винаги трябва да е виртуален, за да могат да се извикат и дестр. на наследниците

Абстрактен клас

def чисто виртуални ф-ии - ф-я без имплементация, предназначена за презаписване от наследниците

```
class Base{
```

```
virtual void f() = 0;
```

```
}
```

Може да има тело, но може да се извика само в деструктор, нивер и в телото на презаписване на ф-я

- не може да се извика от външния свят

def Абстрактен клас - клас с поне една ч.в.ф. => предназначен за наследяване

→ не можем да създаваме обекти от такива класове

→ наследник, който не презаписва ч.в.ф. също е абстрактен

Виртуална таблица - "масив от указатели към ф-ии"

Таблица, в която пише кои ф-ии трябва да се извикат. В всеки клас, съдържащ вирт. ф-ии има своя вирт. таблица

vpointer - всеки обект има виртуален указател, сочещ към вирт. таблица на класа

- той е елемент-данни и заема памет

Base
↑

A(...): Base(...) // Base задава вирт. таблица на A

A

~A() { } // променя вирт. укл. на Base да не е на A
// ~Base() - вие може да се извика ~Base(),
без да се пренасочи към ~A() (тъй като е
Base() { виртуален)

} // тук се задава vtable за A