

# **Objektovo orientované programovanie**

(Programovanie proti rozhraniu)

11. prednáška (2. časť)

Vladislav Novák  
FEI STU v Bratislave  
25.11.2014

## Programovanie proti rozhraniu

V tomto prípade sa pod rozhraním rozumie = programová konštrukcia `interface` + kontrakt.

**Kontrakt** obsahuje ďalšie informácie ktoré nemožno zachytiť programovou konštrukciou `interface`. Tieto informácie sú uvedené v dokumentácii (komentároch). Prekladač dokáže kontrolovať dodržanie programovej konštrukcie `interface`, ale nedokáže kontrolovať dodržanie kontraktu. Niektoré časti kontraktu možno kontrolovať za behu použitím kľúčového slova `assert` (napr. kontrola vstupných a výstupných podmienok metódy, invariantov).

Základom programovania proti rozhraniu je deklarácia premennej ako inštancie rozhrania (konštrukcie `interface`), nie ako inštancie triedy. Ak napríklad chceme v programe použiť zoznam, premenná reprezentujúca zoznam bude typu `List`, nie `ArrayList`, ani `LinkedList`. V prípade potreby je potom ľahké vymeniť implementáciu zoznamu. Nie je vhodné premennú deklarovať ani ako inštanciu niektorej abstraktnej triedy (`AbstractList`, `AbstractCollection`), pretože takáto deklarácia obmedzuje možnosti výmeny implementácie.

Preto je pri používaní objektov ktorých správanie sa môže meniť, vhodné definovať rozhranie. Toto rozhranie potom implementovať triedami a ako typ premenných používať toto rozhranie.