

UNIVERZITA PARDUBICE  
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

POZNÁMKY K PŘEDMĚTU  
PROJEKTOVÁNÍ DATABÁZOVÝCH  
SYSTÉMŮ

MATĚJ TRAKAL  
Poslední úprava: 21. února 2011, 11:28

## **Obsah**

<b>1 Úvodní hodina – 21. 2. 2011</b>	<b>2</b>
--------------------------------------	----------

## 1 Úvodní hodina – 21. 2. 2011

- Vyjmenujte základní principy objektově orientovaného programování:

- zapouzdření
- škálovatelnost
- přednost skládání před děděním
- polymorfismus (mnohotvárnost)
- skrytí implementace
- programování proti rozhraní
- dědičnost

- Vyjmenujte definice následujících pojmů:

**Objekt** instance třídy

**Typ** definuje společnou množinu hodnot, jaké operace s nimi můžeme dělat – stejné jako třída.

**Třída** třída je skupina metod a atributů, které patří k sobě a zapouzdřují funkcionalitu od jednoho prvku.

**Atribut** je datová složka pro ukládání stavu instance objektu/třídy.

**Metoda** je funkce/procedura, která vykonává operace, nebo pomocí ní přistupujeme k atributům objektu.

**Viditelnost** je nastavení práv, které má přistupující k jednotlivým složkám objektu/třídy (veřejný - viditelné, privátní - vidí pouze vlastní třída a potomci třídy, protected - vidí pouze třída, resp. její instance)

**Správa/Zpráva** volání metody

**Abstraktní třída** je třída, ze které nelze vytvořit instanci, je třeba ji podědit a implementovat její funkcionalitu v jiné třídě.

**Rozhraní** je vrstva/třída (totální abstrakce) mezi vnitřní implementací tříd a aplikací, která s těmito složkami pracuje – pokud změníme implementaci, musí rozhraní zůstat zachované, abychom nemuseli přepracovávat aplikační vrstvu. (souvisí úzce se zapouzdřením).

**UML** unified modeling language – je vymodelování toho, co chceme následně naprogramovat. Slouží pro zjednodušený návrh a hlavně pro pochopení, co chceme udělat, dá se v něm vymodelovat v podstatě situace až do funkční realizace SW.

**Metodika** je postup jak se k cíli dobrat, např. právě pomocí UML.

**UP** unified processing – proces vytváření požadavků a postup na vytvoření sw.

- Další pojmy:

**Softwarový systém** datová část, aplikační část, někdy se zahrnují i uživatelé, kteří systém používají.

**Životní cyklus systému** zadání, realizace (požadavky, analýza, návrh, testování, nasazení, údržba), opakování cyklu dokud neskončí životní cyklus systému.

**Princip vývoje systému podle metodiky UP** iterace – miniprojekt – řeší část ucelenou funkcionality projektu. Projetí kroky od požadavky až po nasazení. Řízení rizik požadavků – řešíme nejtěžší částí jako první.

**Fáze vývoje systému podle UP** 4 části – zahájení (cíl: sesumírovat požadavky, případy užití, zvážení rizik, první use case), rozpracování (vytvoření architektury, hrubý usecase, analytické třídy), konstrukce (realizace), zavedení.

**Disciplíny vývoje systému dle UP** tvorba požadavků, návrhy UML – diagramy

**Požadavky dle metodiky UP**

**Z kterých aktivit se skládá fáze „Analýza“ podle metodiky UP** analytické modely (class diagram, sekvenční a komunikační)

**Jaké metody jsou vytvářené ve fázi „Analýza“ podle metodiky UP**

**Popište a definujte jednotlivé prvky diagramu tříd jazyka UML** název třídy, balíček, metody (parametry, návratový typ, viditelnost), atributy (jejich datové typy), vazby mezi třídami a balíčky, násobnosti, rozhraní

## Reference

- [1] INPDS Lach, *Výukový materiál k předmětu Projektování databázových systémů*. 200?.

Části textů jsou doslovně opsány, jiné přepsány vlastními slovy. Tento materiál slouží pouze k výukovým účelům.