

## Aspektovo-orientovaný vývoj softvéru 2016/17

doc. Ing. Valentino Vranić, PhD., ÚISI FIIT STU

Test – 7. november 2016

V simulácii interaktívneho modelu organizácie organizácia pozostáva z zamestnancov zaradených do hierarchicky organizovaných oddelení reprezentovaných triedou `Department`, pričom každé oddelenie má jednoznačné identifikačné číslo vo forme celočíselného atribútu `id`). Zamestnanci – reprezentovaní triedou `Employee` – sa môžu vyskytovať na hocijakej úrovni a majú definovanú činnosť prostredníctvom metódy `perform()`.

Používateľ dáva podnet oddeleniu na realizáciu jeho činnosti vyvolaním metódy `activate()` (abstrahujeme od GUI). Oddelenie potom realizuje túto činnosť prostredníctvom svojich zamestnancov a svojich pododdelení, ktoré zase činnosť prenášajú na svojich zamestnancov a pododdelenia až po poslednú úroveň hierarchie.

V simulácii treba bez zásahu do tried `Department` a `Employee` zabezpečiť, aby zamestnanci ignorovali činnosti, ktorých zdrojom je oddelenie s určitým `id`.

- 1. (2 b)** Uveďte príslušný analytický model v notácii Theme/Doc v pohľade tém a vzťahov (základný pohľad). Transformujte tento pohľad na pohľad pretínajúcich tém.
- 2. (3 b)** Uveďte príslušný návrhový model v notácii Theme/UML. Môžete využiť jej rozšírenie o nepriame vzťahy z notácie JPDD.
- 3. (5 b)** Uveďte kód výsledného aspektu v jazyku AspectJ. Ak by to bolo vhodné, uplatnite pritom aspektovo-orientovaný návrhový vzor Worker Object Creation, Wormhole alebo Cuckoo's Egg. Svoje rozhodnutie zdôvodnite.