

Metódy a prostriedky špecifikácie

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

Skúška – riadny termín – 14. január 2013

A

Priezvisko:	
Meno:	

1b	
2b	
3b	

Skúška trvá 60 minút.

V otázkach 1–13 je len jedna možnosť správna. Vyznačte svoju odpoveď krížikom do tabuľky. Odpovede na otázky 14 a 15 píšete na čistý list, ktorý ste dostali.

	a	b	c	d	e
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					

V prípade opravy jasne vyznačte odpoveď, ktorá platí. Každá správna odpoveď má hodnotu vyznačenú v otázke. Nesprávna odpoveď, vyznačenie viac odpovedí alebo nejednoznačné vyznačenie nebude hodnotené. Postup riešenia sa pre otázky 1–13 nehodnotí. Necelistvý list nebude uznaný.

1. (1 b) Schéma v jazyku Z pozostáva z týchto častí:

- (a) deklarácia a predikát
- (b) nastavenia a zmeny
- (c) podmienky a akcie
- (d) predpoklady, dôsledky a akcie
- (e) deklarácia a akcie

2. (1 b) Modelovanie vlastností umožňuje primárne zachytenie

- (a) štruktúry
- (b) správania
- (c) predpokladov a dôsledkov
- (d) stavov
- (e) variability

3. (1 b) Prechod medzi modelmi v rôznych etapách vývoja softvéru možno vnímať predovšetkým ako proces

- (a) konkretizácie
- (b) špecializácie
- (c) adaptácie
- (d) anihilácie
- (e) abstrakcie

4. (1 b) Metóda OOram je založená primárne na modelovaní

- (a) rozhraní
- (b) objektov
- (c) rolí
- (d) tried
- (e) prípadov použitia

5. (2 b) Je možné realizovať sprístupnenie viacerých prípadov použitia prostredníctvom rôznych formulárov GUI?

- (a) nie
- (b) áno, toto je obvyklé
- (c) áno, ale len s použitím aspektovo-orientovaného programovania
- (d) áno, ak používajú rozdielne ovládacie prvky formulárov
- (e) áno, ale je to nežiaduce

6. (2 b) Na ošetrenie chybového stavu v hlavnom toku prípadu použitia by ste použili

- (a) podmienku
- (b) podtok
- (c) pomocný tok
- (d) alternatívny tok
- (e) výnimku

7. (2 b) Abstraktný prípad použitia nemôže

- (a) byť aktivovaný
- (b) mať toky
- (c) byť spojený s účastníkmi
- (d) mať toky s definovanými krokmi
- (e) byť použitý vo vzťahoch generalizácie

8. (2 b) V jazyku Z sa schémy operácií, ktoré menia a nemenia stav systému,

- (a) nelíšia
- (b) líšia v tom, či môžu alebo nemôžu mať výstup
- (c) líšia v spôsobe použitia sekvencií
- (d) líšia v spôsobe použitia multimnožín
- (e) líšia spôsobom, akým zahŕňajú stav systému

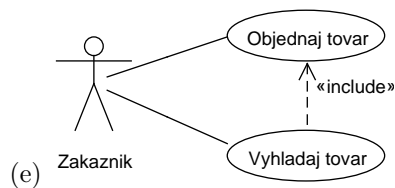
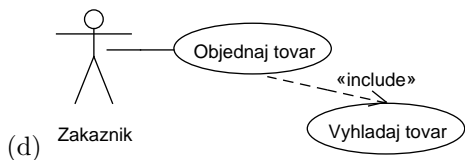
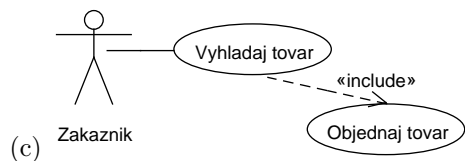
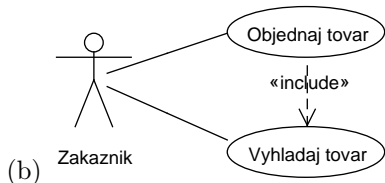
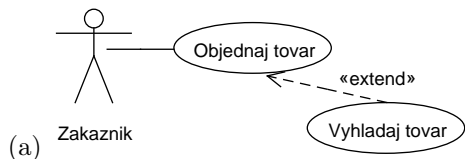
9. (2 b) Abstraktná syntax jazyka UML je definovaná primárne

- (a) jazykom OCL
- (b) diagramami tried
- (c) produkčnými pravidlami gramatiky
- (d) prirodzeným jazykom
- (e) diagramami sekvencií

10. (2 b) V procese vývoja softvéru podľa prístupu Scrum zákazník

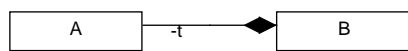
- (a) nemôže v špecifikácií meniť nič
- (b) môže zmeniť najviac 10% špecifikácie
- (c) môže meniť špecifikáciu, ale len ak s tým súhlasí Scrum-Master
- (d) môže meniť špecifikáciu
- (e) môže meniť špecifikáciu, ale len ak s tým súhlasí tím

11. (3 b) V elektronickom obchode medzi obyvkle činnosti zákazníkov patrí vyhľadavanie a prezeranie tovaru a objednávanie tovaru. Zákazník tovar objedná po jeho vyhľadani. Ktorá z nasledujúcich organizácií prípadov použitia by najlepšie zodpovedala opísanej situácii:



12. (3 b) Agregácia na obrázku 1 bude správna

- (a) bez zásahu
- (b) jedine po pridaní atribútu t do triedy A
- (c) jedine po pridaní atribútu t do triedy B
- (d) jedine po zmene kosoštvorca na prázdny
- (e) po zmene kosoštvorca na prázdny alebo po pridaní atribútu t do triedy B



Obr. 1: Diagram k otázke 12.

13. (3 b) V algebraickej špecifikácii grafického objektu okrem iných boli identifikované operácie zistenia farby ($color$) a vyplnenia ($fill$):

$$color : GObject \rightarrow Color$$

$$fill : GObject \times Color \rightarrow GObject$$

Axióm, ktorý hovorí, že po vyplnení ~~krúh~~ grafický objekt má mať zadanú farbu výplne, znie:

(a) $\forall o : GObject \exists c : color \bullet color(o) = c$

- (b) $\exists o : GObject, c : color \bullet fill(o, c) \Rightarrow color(o) = c$
- (c) $\forall o : GObject, c : color \bullet fill(o, c) \Rightarrow color(o) = c$
- (d) $\exists o : GObject, c : color \bullet color(fill(o, c)) = c$
- (e) $\forall o : GObject, c : color \bullet color(fill(o, c)) = c$

14. (5 b) Vyhľadávací stroj aktivuje a spúšťa vyhľadavanie jedným z troch špecializovaných vyhľadávacích strojov na základe dopytu a nastavení. Vyjadrite adekvátnu štruktúru v UML diagramom komponentov. Aký zásah do tejto štruktúry vyžaduje pridanie ďalšieho špecializovaného vyhľadávacieho stroja? (Model dopytu a nastavení nie je predmetom otázky.)

15. (5 b) Udalosť je v osobnom plánovači reprezentovaná triedou $Udalost$. Operácia tejto triedy nastav umožňuje nastaviť parametre udalosti iniciálne a tiež aj ich následnú zmenu:

```
nastav(zaciatok : Date,
       koniec : Date, opis : String)
```

kde $Date$ predstavuje dátum a čas. Pritom nie je povolené meniť začiatok a koniec udalosti, ktorá prebieha alebo skončila, čo je vyjadrené zodpovedajúcim ohraničením v OCL:

```
context Udalost::
  nastav(zaciatok : Date,
         koniec : Date, opis : String) : void
  pre zmenaCasu:
  — zaciatok.before(teraz) and
  — koniec.before(teraz)
  teraz.before(zaciatok) and
  teraz.before(koniec)
```

kde operácia $before()$ triedy $Date$ vracia true, ak je daný dátum a čas pred dátumom a časom poskytnutým ako parameter, a inak vracia false.

Od triedy $Udalost$ je odvodená trieda $ExternaUdalost$, ktorá predstavuje udalosť prevzatú z externého zdroja. Časové údaje tohto typu udalosti sa po iniciálnom nastavení, pred ktorým majú hodnotu $null$, nesmú meniť. Ako toto dosiahnuť v jazyku OCL vzhľadom na existenciu ohraničenia $zmenaCasu$? Bolo by to korektné z hľadiska možného použitia týchto tried?

Metódy a prostriedky špecifikácie

doc. Ing. Valentino Vranić, PhD., ÚISI FIIT STU
Skúška – riadny termín – 14. január 2013

A

35 b

1 a

2 e

3 a

4 c

5 b

6 d

7 a

8 e

9 b

10 d

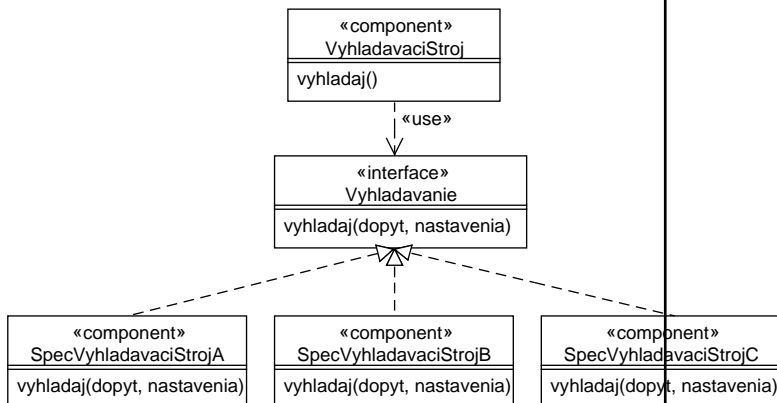
11 b

12 a

13 e, d¹

14

Vhodná štruktúra je na zobrazená nižšie:



Špecializované vyhľadávania sú zastrešené rozhraním *Vyhľadavanie*. Základný vyhľadávací stroj k nim potom môže pristupovať jednotne. (Rozhodovanie by mohlo prebehnúť na základe typu dopytu a nastavení, na čo by bol vhodný (dvojitý) Visitor. Toto je skryté v komponente *VyhladavaciStroj* a nebolo predmetom otázky.)

Pridanie ďalšieho špecializovaného vyhľadávacieho stroja bude znamenať rozšírenie uvedenej štruktúry o ďalší špecializovaný vyhľadávací stroj, ktorý tiež bude realizovať rozhranie *Vyhľadavanie*. (Môže byť potrebný zásah do operácie *vyhladaj()* komponentu *VyhladavaciStroj*, čo však nie je predmetom otázky.)

15

V tejto otázke sa vyskytla menšia chyba. Vo volaní operácie *before()* boli vymenené identifikátory *zaciatok* a *teraz*, ako aj *koniec* a *teraz*, takže podmienka *zmenaCasu* v skutočnosti kontrolovala, či udalosť už neskončila, namiesto, aby kontrolovala, či udalosť už nezačala. Vysvetlenie však bolo správne a podstata otázky v inej veci.

Na to, aby v triede *ExternaUdalost* odvodenej od triedy *Udalost* bola zmenená podmienka prekonanej operácie *nastav()* treba túto podmienku prekonať:

```
context ExternaUdalost::
    nastav(zaciatok : Date ,
    koniec : Date , opis : String) : void
pre zmenaCasu:
    zaciatok = null and
    koniec = null
```

Toto však predstavuje zosilnenie predpokladu, čo predstavuje porušenie Liskovej princípu substitúcie. Dôsledok je ten, že by klientsky kód, ktorý by (správne) vychádzal z predpokladu prekonanej operácie, mohol (nesprávne) zmeniť aj časové údaje externej udalosti.

¹V otázke 13 namiesto *grafický objekt* bol mylne uvedený *kruh*. Preto je ako správna uznávaná aj odpoveď d (s existenčným kvantifikátorom namiesto všeobecného), hoci v skutočnosti by nebola správna, ani keby v otázke zostal *kruh*.