Objektovo-orientované programovanie **A**

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

Semestrálny test – 28. marec 2012

Priezvisko:

Meno:

1b	
2b	

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

Test trvá 35 minút. V otázkach je len jedna možnosť správna. Vyznačte svoju odpoveď krížikom do tabuľky. Hodnotia sa len odpovede v tabuľke. V prípade opravy jasne vyznačte odpoveď, ktorá platí. Každá správna odpoveď má hodnotu vyznačenú v otázke. Nesprávna odpoveď, vyznačenie viac odpovedí alebo nejednoznačné vyznačenie má hodnotu 0 bodov. Postup riešenia sa nehodnotí. Len celistvý list bude akceptovaný.

1. (1 b) Inštancia rozhrania v Jave

(a) je abstraktná
(b) je statická
(c) je generická
(d) je polymorfná
(e) nejestvuje

2. (1 b) Idióm Javy, v ktorom výber metódy závisí od dvoch príjemcov sa volá

(a) double trouble
(b) double select
(c) double dispatch
(d) double mismatch
(e) double method

3. (2 b) Daný je nasledujúci kód v Jave (každá trieda vo vlastnom súbore):

```
public class A {
   public void f() {
      System.out.print("A");
   }
   public void af() {
      System.out.print("Af");
   }
}

public class B extends A {
   public void f() {
      System.out.print("B");
   }
   public void bf() {
      System.out.print("Bf");
   }
}
```

Potom tento kód

```
A a = new B();
B b = new B();

a.f();
a.af();
((B) a).bf();
(new B()).f();
(new B()).bf();
(new B()).af();
```

vypíše:

(a) nič
(b) AAfBfABf
(c) AAfBfABfAf
(d) BAfBfBBfAf
(e) BAfBfABfAf

4. (1 b) Prístup Design by Contract je založený na

(a) princípe otvorenosti a uzávretosti
(b) princípe, že návrhu musí predchádzať zmluva
(c) dôslednom ošetrovaní výnimiek
(d) princípe, že implementácii musí predchádzať návrh
(e) Liskovej princípe substitúcie

5. (1 b) Balíky v Jave sa definujú

(a) označovaním súborov so zdrojovým kódom názvami balíkov
(b) zaradením súborov so zdrojovým kódom do hierarchie adresárov, ktorá zodpovedá želanej hierarchii balíkov
(c) zaradením zdrojového kódu tried do súborov, ktoré reprezentujú balíky
(d) zaradením súborov so skompilovaným kódom do hierarchie adresárov, ktorá zodpovedá želanej hierarchii balíkov
(e) importovaním tried

6. (1 b) To, že sa v Jave veľkosť poľa dá zadať aj takto (ak metóda `m()` vracia **int**)

```
X[] a = new X[new C().m()];
```

(a) je špeciálnou vlastnosťou Javy, ktorá nesúvisí s objektmi
(b) je dôsledkom optimalizácie
(c) je dôsledkom toho, že pole v Jave je objekt
(d) je dôsledkom novších rozšírení Javy
(e) je spôsobené vyššou dostupnosťou pamäte

7. (2 b) Daný je nasledujúci kód:

```
List<Z> z = new LinkedList<>();
z.add(new X());
z.add(new Y());
```

Aby sa tento kód mohol preložiť a vykonať

(a) `Z` musí byť rozhranie, a typy `X` a `Y` triedy, ktoré ho implementujú
(b) typy `X` a `Y` musia byť triedy priamo alebo nepriamo odvodené od typu `Z`
(c) `Z` musí byť abstraktná trieda, a typy `X` a `Y` triedy od nej odvodené
(d) typy `X` a `Y` musia byť triedy odvodené od typu `Z`, ale `X` nesmie byť odvodené od `Y` alebo naopak
(e) typ `X` musí byť odvodený od typu `Y` alebo naopak

8. (1 b) Daný je nasledujúci kód v Jave:

```
public class A {
   void m(int i) { }
   void m(int i, int j) { }
}
public class B extends A {
   void n(int i) { }
   void m(int i, int j) { }
}
```

(a) `B.m(int i, int j)` preťažuje `A.m(int i)`
(b) `B.n(int i)` prekonáva `A.m(int i)`
(c) `A.m(int i, int j)` prekonáva `A.m(int i)`
(d) `B.m(int i, int j)` preťažuje `B.n(int i)`
(e) `B.n(int i)` preťažuje `A.m(int i)`

9. (1 b) Daný je nasledujúci kód v Jave:

```
class Utvar {
   int farba = 12;
   void nakresli() { }
}

class Kruh extends Utvar {
   void nakresli(int f) {
      System.out.println("Kruh farby " + f);
   }
}
```

Potom kód:

```
Kruh k = new Kruh();
k.nakresli();
```

(a) spôsobí chybu pri preklade
(b) vypíše text „Kruh farby 12“
(c) nevypíše nič
(d) spôsobí chybu počas vykonávania
(e) vypíše text „Kruh farby“

10. (1 b) Explicitne vytvorený konštruktor triedy v konštruktore odvodenej triedy v Jave

(a) nemusí byť zavolaný
(b) nemôže byť zavolaný
(c) je lepšie, aby nebol zavolaný
(d) nemá význam volať
(e) musí byť zavolaný

11. (2 b) Daný je nasledujúci kód v Jave:

```
public class A {
   private A() { }
   private static A a = new A();
   public static A m() {
      return a;
   }
   public void f() {
      System.out.println("OK");
   }
}
```

Výpis hlásenia „OK“ z inej triedy možno dosiahnuť

(a) pomocou **new** `A().f()`
(b) aj pomocou **new** `A().f()`, aj pomocou `A.m().f()`;
(c) pomocou `A.f().m()`;
(d) `A.m().f()`;
(e) aj pomocou **new** `A().f()`, aj pomocou `A.f().m()`;

12. (1 b) Zachytenú výnimku v Jave je možné

(a) len ošetriť
(b) len zhltnúť
(c) len vypísať
(d) znovu vyhodiť
(e) len vypísať a/alebo ošetriť

Objektovo-orientované programovanie **A**

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

Semestrálny test – 28. marec 2012

15 b

1 e

2 c

3 d

4 e

5 a

6 c

7 b

8 a

9 c

10 e

11 d

12 d