

# Objektovo-orientované programovanie

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

Test – 2. apríl 2014

**B**

<b>Priezvisko:</b>	tlačeným písmom
<b>Meno:</b>	

1b	
2b	

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	

Test trvá 30 minút.

Odpovede na otázky vpíšte do tabuľky. Hodnotia sa len odpovede v tabuľke. Otázky sú s výberom odpovede, z ktorých je len jedna možnosť správna.

V prípade opravy jasne vyznačte odpoveď, ktorá platí. Každá správna odpoveď má hodnotu vyznačenú v otázke. Nesprávna, nejednoznačná alebo neúplná odpoveď má hodnotu 0 bodov. Postup riešenia sa nehodnotí.

Poškodený list nebude uznaný.

## 1. (1 b) Zavedenie viacerých balíkov v Jave môže spôsobiť

- (a) riešiteľnú kolíziu názvov zavedených prvkov
- (b) neriešiteľnú kolíziu názvov zavedených prvkov
- (c) zvýšenie veľkosti class súborov tried, ktoré používajú prvky príslušného balíka
- (d) zvýšenie veľkosti výsledného JAR súboru
- (e) spomalenie programu

## 2. (2 b) Pokus o preklad a vykonanie nasledujúcej triedy v Jave:

```
public class A {  
    public static int f(int i) {  
        return 2 * i;  
    }  
    public static void main(String[] args) {  
        int[] a = new int[f(5)];  
    }  
}
```

- (a) prekladáč hlási chybu pre nekonzistenciu typov
- (b) prekladáč hlási chybu pre neznámu veľkosť poľa **a**
- (c) pri vykonávaní vznikne výnimka pre neznámu veľkosť poľa **a**
- (d) prebehne korektne
- (e) pri vykonávaní vznikne výnimka pre nekonzistenciu typov

## 3. (1 b) Modifikátory prístupu v Jave umožňujú

- (a) redukovať počet atribútov
- (b) regulovať, či iní môžu vidieť kód a zasahovať doň
- (c) redukovať závislosti v kóde
- (d) redukovať počet metód
- (e) regulovať, či iní môžu vidieť kód

## 4. (1 b) Koľko inštancií môže mať anonymná trieda?

- (a) má vždy práve jednu
- (b) ani jednu
- (c) nemusí mať ani jednu, ale môže ich mať viac
- (d) musí mať aspoň jednu, ale môže ich mať viac
- (e) nemusí mať ani jednu, ale môže mať jednu

## 5. (1 b) Inštancia abstraktnej triedy v Jave

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

## 6. (1 b) V objektovo-orientovanom programe hlavná funkcionálna typicky

- (a) je obsiahnutá v konštruktoroch
- (b) vzniká dedením
- (c) je obsiahnutá v metóde `main()`
- (d) je zabezpečená statickými metódami
- (e) vzniká v interakciách objektov

## 7. (1 b) Dedenie v objektovo-orientovanom programovaní treba aplikovať

- (a) len pri anonymných triedach
- (b) berúc do úvahy implicitné predpoklady a dôsledky
- (c) čím menej
- (d) berúc do úvahy implicitné správanie a štruktúru
- (e) čím viac

## 8. (1 b) Iterátory v Java API uľahčujú

- (a) opakovanie vykonávania ľubovoľného kódu
- (b) rušenie prvkov v zoskupeniach
- (c) volania abstraktných metód
- (d) prechádzanie zoskupeniami
- (e) pridávanie prvkov do zoskupení

9. (2b) Daný je kód v Jave na obr. 1. Čo sa vypíše vykonaním triedy M?

- (a) a.sp.p.sp.b.sp.p.sp.
- (b) a.sa.p.sp.b.sb.p.sp.
- (c) a.sa.a.sp.b.sb.b.sp.
- (d) a.sp.a.sp.b.sp.b.sp.
- (e) a.sa.a.sa.b.sb.b.sb.

```
class P {
    public void m() { System.out.print("p."); }
    public static void s() {
        System.out.print("sp.");
    }
}
class A extends P {
    public void m() { System.out.print("a."); }
    public static void s() {
        System.out.print("sa.");
    }
}
class B extends P {
    public void m() { System.out.print("b."); }
    public static void s() {
        System.out.print("sb.");
    }
}
class M {
    public static void op(P o) {
        o.m();
        o.s();
    }
    public static void main(String[] args) {
        A o1 = new A();
        P o2 = new B();

        M.op(o1);
        M.op((P)o1);
        M.op(o2);
        M.op((P)o2);
    }
}
```

Obr. 1: Kód pre otázky 9 a 12.

10. (1b) Typ výnimky v Jave je určený jej

- (a) krátkym opisom
- (b) identifikačným číslom
- (c) triedou
- (d) metódami
- (e) názvom

11. (1b) V programe v Jave je daná trieda `Objednavka`. Výraz `Objednavka.class` predstavuje referenciu na

- (a) objekt, ktorý reprezentuje triedu `Objednavka`
- (b) názov triedy `Objednavka` typu `String`
- (c) atribút triedy `Objednavka`
- (d) súbor, ktorý reprezentuje triedu `Objednavka`
- (e) inštanciu triedy `Objednavka`

12. (2b) K triedam na obr. 1 je daný nasledujúci kód v Jave:

```
((B)o1).m();
```

Tento kód

- (a) sa vykoná úspešne a nevypíše nič
- (b) sa nebude dať preložiť
- (c) sa vykoná úspešne a vypíše a.
- (d) nevypíše nič a vyhodí výnimku
- (e) sa vykoná úspešne a vypíše b.

15 b

1 a

2 d

3 c

4 d – bude akceptovaná aj odpoveď a

5 d

6 e

7 b

8 d

9 d

10 c

11 a

12 b