

Priezvisko:		Body
Meno:		

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

Test trvá 50 minút.

V otázkach je len jedna možnosť správna. Vyznačte svoju odpoveď krížikom do veľkej tabuľky (malú tabuľku nevyplňajte). Hodnotia sa len odpovede vyznačené v tabuľke.

V prípade opravy jasne vyznačte odpoveď ktorú vyberáte. Každá správna odpoveď má hodnotu vyznačenú v otázke. Nesprávna odpoveď, vyznačenie viac odpovedí alebo nejednoznačné vyznačenie má hodnotu 0 bodov. Postup riešenia sa nehodnotí.

1. (1 b) V programe v Java sa príkaz

```
package my;
```

- (a) môže vyskytovať vo viacerých súboroch toho istého programu len ak sú v tom istom adresári
- (b) nemôže vyskytovať v súboroch
- (c) môže vyskytovať len v jednom súbore
- (d) môže vyskytovať vo viacerých súboroch toho istého programu
- (e) môže vyskytovať len v súbore ktorý obsahuje len jednu triedu alebo rozhranie

2. (1 b) Nech `n` je atribút triedy typu `int`. Refazec `r` je v jednej z metód tej istej triedy definovaný takto:

```
Integer r[] = new Integer[n];
```

Táto definícia je v Java

- (a) nekorektná
- (b) korektná
- (c) korektná jedine ak je atribút `n` statický
- (d) korektná jedine ak je atribút `n` finálny
- (e) korektná jedine ak atribút `n` nie je inicializovaný metódou

3. (1 b) Abstraktná trieda v Java

- (a) môže mať len abstraktné metódy
- (b) nemôže mať metódy
- (c) nemôže mať inicializované atribúty
- (d) nemôže mať prekonávajúce metódy
- (e) nemusí mať len abstraktné metódy

4. (1 b) Daný je nasledujúci kód v Java obsiahnutý v jednej z metód:

```
abstract class A implements I { }  
interface I {  
    void m();  
}  
A[] a = new I[5];
```

Tento kód sa

- (a) nepreloží, lebo nie je možné definovať triedy a rozhrania v metódach
- (b) nepreloží, lebo posledný riadok je zlý
- (c) preloží a vykoná korektné
- (d) preloží, ale vznikne chyba pri vykonávaní posledného riadku
- (e) nepreloží, lebo trieda `A` neimplementuje metódu `m()`

5. (1 b) Trieda v objektovo-orientovanom programovaní predstavuje

- (a) inštanciu
- (b) objekt
- (c) rozhranie objektu
- (d) typ objektu
- (e) interakciu objektov

6. (1 b) Prístup `private` je vhodné použiť pri takých prvkoch triedy, ku ktorým chceme pristupovať len

- (a) v odvodených triedach a v triedach toho istého balíka
- (b) v danej triede
- (c) v odvodených triedach
- (d) v triedach toho istého balíka
- (e) v odvodených triedach toho istého balíka

7. (2 b) Daný je nasledujúci program v Java:

```
class C {  
    C x;  
    void m() {  
        x = new SubC();  
        **1**.f();  
    }  
    public static void main(String[] args) {  
        **2**.m();  
    }  
}  
  
class SubC **3** {  
    void f() {  
        System.out.println("f");  
    }  
}
```

Ktoré fragmenty kódu treba v tomto programe doplniť, aby sa pri jeho vykonaní vypísalo `f`?

- (a) `**1**:` `x` `**2**:` `new C()`
 `**3**:` `extends C`
- (b) `**1**:` `x` `**2**:` `new C()`
 `**3**:` `implements C`
- (c) `**1**:` `((SubC)x)` `**2**:` `new C()`
 `**3**:` nič
- (d) `**1**:` `((SubC)x)` `**2**:` `new C()`
 `**3**:` `extends C`
- (e) `**1**:` `SubC` `**2**:` `C`
 `**3**:` `extends C`

8. (2 b) Daný je kód v Jave na obr. 1. Čo sa vypíše po vykonaní týchto príkazov:

```
I o = new B();  
o.m();  
((A)o).m();  
((I)o).m();
```

- (a) bba
- (b) aaa
- (c) baa
- (d) bbb
- (e) bab

```
interface I {  
    void m();  
}  
  
class A implements I {  
    public void m() { System.out.print("a"); }  
}  
  
class B extends A {  
    public void m() { System.out.print("b"); }  
}
```

Obrázok 1: Kód pre otázky 8, 9 a 11.

9. (2 b) K triedam z obr. 1 je daný nasledujúci kód:

```
List<I> z = new ArrayList<A>();  
z.add(new B());
```

Tento kód sa

- (a) nepreloží, lebo typ referencie z nezodpovedá typu priradeného objektu
- (b) nepreloží, lebo do zoznamu z sa dajú vkladať len objekty typu A
- (c) preloží a vykoná korektne
- (d) preloží, ale padne počas vykonávania s výnimkou `ClassCastException`
- (e) nepreloží, lebo trieda `ArrayList` nie je generická

10. (1 b) Liskovej princíp substitúcie

- (a) hovorí o zapuzdrení
- (b) predstavuje kritérium pre použitie dedenia
- (c) stanovuje kritéria pre použitie abstraktných tried
- (d) odčleňuje správanie objektu od jeho implementácie
- (e) predstavuje kritérium pre použitie agregácie

11. (1 b) Daný je kód v Jave na obr. 1. Dá sa z metódy `m()` triedy B zavolať rovnomenná metóda triedy A?

- (a) áno, príkazom `m()`;
- (b) nie
- (c) áno, príkazom `A.m()`;
- (d) áno, príkazom `this.m()`;
- (e) áno, príkazom `super.m()`;

12. (1 b) Dá sa v Jave urobiť inštancia rozhrania?

- (a) áno
- (b) nie
- (c) áno, ale len ak neobsahuje metódy
- (d) áno, ale len ak obsahuje výlučne statické prvky
- (e) áno, ale nebudú sa dať zavolať metódy

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