



Priezvisko:	
Meno:	

1 b	
2 b	
3 b	

Skúška trvá 120 minút.

V otázkach 1–16 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 ktorú odpoveď vyberáte. Každá správna odpoveď má hodnotu vyznačenú v otázke. Nesprávna odpoveď, vyznačenie viac odpovedí alebo nejednoznačné vyznačenie má hodnotu 0 bodov. Postup riešenia sa pre otázky 1–16 nehodnotí.

Odpovede na otázky 17 a 18 píšete na prídavný list. Na ňom tiež uveďte svoje priezvisko a meno.

	a	b	c	d	e
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					

1. (1 b) Príkaz

```
import java.util.*;
```

- (a) sprístupní priestor názvov všetkých typov balíka java.util vrátane typov v podbalíkoch
- (b) sprístupní priestor názvov všetkých typov balíka java.util, ale bez typov v podbalíkoch
- (c) fyzicky pripojí typy balíka java.util k programu bez typov v podbalíkoch
- (d) fyzicky pripojí len skutočne použité typy balíka java.util k programu bez typov v podbalíkoch
- (e) fyzicky pripojí všetky typy balíka java.util k programu vrátane typov v podbalíkoch

2. (1 b) Na rozdiel od Javy jazyk C++

- (a) automaticky vytvára objekty
- (b) automaticky ruší nereferencované objekty
- (c) má explicitné konštruktory
- (d) má explicitné deštruktory
- (e) má implicitné konštruktory

3. (1 b) Abstraktná metóda v Jave

- (a) nemôže mať argumenty
- (b) môže mať telo
- (c) nemôže mať návratovú hodnotu
- (d) môže sa vyskytovať v hocijakej triede
- (e) nemôže byť statická

4. (1 b) Pole triedy, ktorému predchádza kľúčové slovo private

- (a) je dostupné len v danej triede
- (b) je dostupné len v rámci jednej triedy
- (c) sa nezapíše do súboru pri serializácii objektu
- (d) je dostupné len v danej hierarchii tried
- (e) je chránené pred zápisom

5. (2 b) Účelom aspektov v aspektovo-orientovanom programovaní je

- (a) urychlenie vykonávanie programu
- (b) definovanie rozdelenie kódu do komponentov
- (c) definovanie miesta vo vykonávaní programu
- (d) zapletenie a roztrúsenie kód
- (e) modifikácia štruktúru a vykonávanie programu

6. (2 b) Nech o je objekt triedy ktorá poskytuje metódu int m(). Reťazec r je definovaný takto:

```
Integer r[] = new Integer[o.m()];
```

Táto definícia je

- (a) nekorektná
- (b) korektná
- (c) korektná jedine ak je metóda m() finálna
- (d) korektná jedine ak je metóda m() statická
- (e) korektná jedine ak je metóda m() synchronizovaná

7. (2 b) Najdôležitejšie kritérium pre použitie dedenia je

- (a) miera použitia abstraktných tried
- (b) existencia vzťahu typ-podtyp
- (c) zachovanie zapuzdrenia
- (d) znovupoužitie polí a metód nadtriedy v podtriedach
- (e) agregácia tried

8. (2 b) V jazyku C++ ste implementovali komplexné čísla vlastnou triedou. Na zjednodušenie zápisu matematických operácií s komplexnými číslami je vhodné použiť

- (a) statické funkcie
- (b) preťaženie funkcií
- (c) preťaženie operátorov
- (d) virtuálne funkcie
- (e) virtuálne operátory

9. (2 b) V diagrame sekvencií jazyka UML horizontálne šípky s plnou čiarou označujú

- (a) zľava doprava vyvolanie operácie, a zprava doľava návrat hodnoty
- (b) návrat hodnoty
- (c) tok údajov medzi objektmi
- (d) vyvolanie operácie
- (e) prenos parametrov operácie v smere šípky

10. (2 b) V jazyku AspectJ bodové prierezy (pointcuts) slúžia na

- (a) modifikáciu vykonávania programu
- (b) definovanie nových metód
- (c) doplnenie nových prvkov do tried
- (d) pretínanie tried
- (e) zachytenie bodov spájania

11. (3 b) Návrhový vzor Visitor slúži na

- (a) pridávanie vzťahov medzi triedami bez ich zmeny
- (b) zabránenie vytvárania viac než jednej inštancie danej triedy
- (c) pridávanie operácií nad objektmi daných tried bez ich zmeny
- (d) definovanie závislosti stavu viacerých objektov od ďalšieho objektu
- (e) zabránenie rozširovania kódu

12. (3 b) Daný je program v Jave pozostávajúci z dvoch balíkov. V jednom z nich je nasledujúci kód:

```
package balik1;
```

```
class Trieda1 {  
    private int a;  
    protected int b;  
    String c;  
}
```

```
public class Trieda2 extends Trieda1 {  
    public String d;  
    public void m1() {  
        a = 1; // 1  
        b = 2; // 2  
        c = "a"; // 3  
    }  
}
```

Druhý balík obsahuje tento kód:

```
package balik2;  
import balik1;
```

```
public class Trieda3 {  
    public static void main(String[] args) {  
        Trieda2 t = new Trieda2();  
  
        t.b = 1; // 4  
        t.c = "a"; // 5  
        t.d = "b"; // 6  
    }  
}
```

Všetky príkazy označené komentárom ktoré porušujú prístupnosť prvkov tried sú:

- (a) 1, 4 a 5
- (b) 1
- (c) 1, 2, 4 a 5
- (d) 1, 4, 5 a 6
- (e) 1 a 4

13. (3 b) K triedam z obr. 1 je daný nasledujúci kód (s importovaným balíkom java.util):

```
List<C> list = new ArrayList<X>();  
list.add(new X());
```

Tento kód sa

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

14. (3 b) Daný je kód v Jave na obr. 1. Čo sa vypíše po vykonaní týchto príkazov:

```
C o = new C();  
o.s();  
o = new X();  
o.s();  
o.m();
```

- (a) C.s() X.s() X.m()

```
class C {  
    void m() { System.out.print("C.m() "); }  
    static void s() { System.out.print("C.s() "); }  
}
```

```
class X extends C {  
    void m() { System.out.print("X.m() "); }  
    static void s() { System.out.print("X.s() "); }  
}
```

Obrázok 1: Kód pre otázky 13, 14 a 15.

- (b) C.s() C.s() X.m()
- (c) X.s() X.s() X.m()
- (d) C.s() C.s() C.m()
- (e) C.s() C.s() X.m()

15. (3 b) Daný je kód v Jave na obr. 1. Dá sa z metódy s() triedy X zavolať metóda m() triedy X bez vytvorenia objektu triedy X?

- (a) nie
- (b) áno, príkazom X.m();
- (c) áno, príkazom m();
- (d) áno, príkazom super.m();
- (e) áno, príkazmi X o; o.m();

16. (3 b) Na sledovanie kliknutia na tlačidlo v rámci Swing by ste

- (a) vytvorili niť udalosti a registrovali ju pre tlačidlo
- (b) vytvorili prijímač udalosti a registrovali ho pre tlačidlo
- (c) v ďalšej niti sledovali v slučke, či používateľ neklikol keď kurzor bol na tlačidle
- (d) registrovali tlačidlo v niti na odosielanie udalosti
- (e) použili rozloženie FlowLayout

17. (4 b) Nakreslite diagram tried v jazyku UML, ktorý zodpovedá nasledujúcemu kódu v Jave:

```
class M {  
    M m;  
    A[] a;  
}  
  
class B extends A implements I {  
}
```

Agregáciu reprezentujte graficky (hranou).

18. (12 b) V aplikácii sa pracuje s dokumentmi vo formáte PS, PDF a BMP, ktoré je možné aj vytlačiť. Všeobecný ovládač tlačiarne radu XPrint poskytuje operácie pre tlač dokumentov v každom z formátov, ktoré podporuje aplikácia. Hoci je tento ovládač možné použiť pre všetky tlačiarne radu XPrint, lepšie výsledky sa dosahujú použitím zodpovedajúcich ovládačov pre jednotlivé typy tlačiarne XPrintA a XPrintB. Ovládače tlačia dokumenty v poradí v akom ich dostali.

Napište relevantný kód v Jave, ktorý zodpovedá týmto požiadavkám, využívajúc pritom mechanizmy objektovo-orientovaného programovania v maximálnej miere. Kód vysvetlite. Nie je potrebné, aby ste uvádzali skutočný kód pre spracovanie dokumentov.



1 b

2 d

3 e

4 a
—

5 e

6 b

7 b

8 c

9 d

10 e
—

11 c

12 a

13 c

14 b /e

15 a

16 b

50