

Priezvisko:	tlačeným písmom
Meno:	

1b	
2b	

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

Test trvá 30 minút.

Odpovede na otázky vpíšte do tabuľky. Hodnotia sa len odpovede v tabuľke. Pri otázkach s výberom odpovede 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) Ktorý návrhový vzor implementuje tento kód v Jave (každá trieda a rozhranie vo vlastnom súbore)?

```
public interface K {
    void m(L o);
}

public interface L {
    void f(C e);
    void f(D e);
}

public class C implements K {
    ...
    public void m(L o) {
        o.f(this);
    }
}

public class D implements K {
    ...
    public void m(L o) {
        o.f(this);
    }
}

public class M implements L {
    public void f(C e) {
        ...
    }
    public void f(D e) {
        ...
    }
}

public class N implements L {
    public void f(C e) {
        ...
    }
    public void f(D e) {
        ...
    }
}
```

- (a) Composite
- (b) Visitor
- (c) žiaden
- (d) Strategy
- (e) Observer

2. (1 b) Niť v Jave vzniká

- (a) automaticky z každej metódy
- (b) automaticky pre každú triedu
- (c) využitím príslušného mechanizmu Java API
- (d) procesom opačným k serializácii
- (e) pre každú metódu označenú kľúčovým slovom **synchronized**

3. (1 b) Okrem iného, program zabezpečuje výpočet podľa určitého vzorca. Používateľ tento výpočet nakoniec spúšťa kliknutím na tlačidlo v používateľskom rozhraní realizovanom v rámci typu JavaFX alebo Swing. Samotný výpočet by z hľadiska objektovo-orientovaného návrhu bolo najlepšie realizovať

- (a) v zodpovedajúcej triede aplikačnej logiky
- (b) v metóde main()
- (c) ako metódu okna, v ktorom sa nachádza tlačidlo
- (d) v implementácii spracovateľa/prijímača (event handler/listener) tlačidla
- (e) ako metódu tlačidla

4. (1 b) Daná je trieda

```
public class C implements Serializable {
    public String id;
    public List<C> l = new ArrayList<>();

    public C(String id) {
        this.id = id;
    }

    public static void main(String[] args)
        throws ClassNotFoundException, IOException {

        C a = new C("a");
        C b = new C("b");
        C c = new C("c");
        C d = new C("d");

        b.l.add(a);
        b.l.add(d);
        c.l.add(a);
        c.l.add(b);
        c.l.add(d);
        d.l.add(a);

        ObjectOutputStream out =
            new ObjectOutputStream(
                new FileOutputStream("f.out"));
        out.writeObject(b);
        out.close();
    }
}
```

Ktoré z objektov a, b, c a d sa zaznamenajú do súboru f.out (uveďte príslušné písmená; na poradí nezáleží)?

5. (1 b) Zavedenie (import) viacerých balíkov v Jave môže spôsobiť

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

6. (1 b) Zahŕnutie objektu jednej triedy v objekte inej triedy sa v objektovo-orientovanom programovaní označuje ako

- (a) dedenie
- (b) agregácia
- (c) integrácia
- (d) segregácia
- (e) generickosť

7. (1 b) Spracovanie výnimky v Jave sa realizuje

- (a) blokom kódu spracovania na mieste jej vyhodnenia
- (b) implementáciou metódy `try()` v triede výnimky
- (c) v metóde `main()`
- (d) implementáciou metódy `catch()` v triede výnimky
- (e) blokom kódu spracovania na mieste jej zachytenia

8. (1 b) Anonymné triedy, odkazy na metódy a lambda výrazy sa v Jave používajú na

- (a) prenos funkcionality
- (b) zrýchlenie programu
- (c) zmenšenie programu
- (d) zvýšenie bezpečnosti
- (e) šetrenie pamäťou

9. (1 b) Atribút triedy, ktorému predchádza kľúčové slovo **private**

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

10. (1 b) Čo sa vypíše po spustení nasledujúceho programu v Jave?

```
interface E {
    void m();
}
abstract class F implements E {
    public void m() {
        System.out.print("F");
    }
}
class G extends F {
    public void m() {
        super.m();
        System.out.print("G");
    }
}
class H extends G {
    public void m() {
        super.m();
        System.out.print("H");
    }
}
class M {
    public static void exe(E... o) {
        for (E e : o) {
            e.m();
            System.out.print("-");
        }
    }
    public static void main(String[] args) {
        E e1 = new H();
        G e2 = new G();
        F e3 = new G();
        E e4 = (G) new H();

        exe((G) e1, e2, e3, e4);
    }
}
```

Objektovo-orientované programovanie 2017/18

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

Test – 27. marec 2018

A

10 b

1 b

2 c

3 a

4 a, b, d

5 c

6 b

7 e

8 a

9 d

10 FGH-FG-FG-FGH-