

(vyplňte tlačítkom)

Priezvisko:**Meno:**

1 b	
2 b	
3 b	

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

Skúška trvá 70 minút.

Odpovede na otázky 1–12 vpíšte do tabuľky. Pri týchto otázkach sa hodnotia len odpovede v tabuľke (bez postupu). Odpoveď musí byť jednoznačná a čitateľná, inak má hodnotu 0 bodov.

V otázkach s ponúknutými odpoveďami je len jedna možnosť správna – do tabuľky vpíšte len písmeno, ktorým je označená odpoveď, ktorú vyberáte.

Odpoveď na otázku 13 píšete výlučne na list, na ktorom sa nachádza jej znenie.

Poškodený list nebude uznaný.

1. (1 b) Analogickým mechanizmom k šablónam (templates) v jazyku C++ v Jave

- (a) sú anonymné triedy
- (b) je generickosť
- (c) nejestvuje
- (d) sú rozhrania
- (e) RTTI

2. (1 b) Triedy v jazyku C++ majú pôvod v mechanizme

- (a) union
- (b) template
- (c) define
- (d) struct
- (e) include

3. (1 b) V jazyku C# udalosti

- (a) jestvujú ako mechanizmus jazyka, ale nemožno ich použiť mimo GUI
- (b) jestvujú ako mechanizmus jazyka a možno ich použiť aj mimo GUI
- (c) nemožno implementovať
- (d) sú to isté čo mechanizmus delegátov
- (e) nepredstavujú mechanizmus jazyka

4. (1 b) Java podporuje perzistenciu prostredníctvom

- (a) agregácie
- (b) radializácie
- (c) modularizácie
- (d) synchronizácie
- (e) serializácie

5. (1 b) Je možné len pomocou medzitypových deklarácií (bez použitia RTTI) v jazyku AspectJ ovplyvniť správanie programu?

- (a) áno
- (b) áno, ale len nestatických metód
- (c) nie
- (d) áno, ale nie nastavovanie atribútov
- (e) áno, ale len statických metód

6. (1 b) Generickosť v Jave umožňuje, aby zoskupenia

- (a) mohli byť špecializované pre akýkoľvek typ údajov pri de- dení
- (b) fungovali rýchlejšie
- (c) mohli uchovať väčší počet objektov
- (d) automaticky ukladané na disk
- (e) mohli byť špecializované pre akýkoľvek typ údajov pri inštanciacii

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

```
class A extends Thread {
    C c;
    public A(C c) {
        this.c = c;
    }
    public void run() {
        for (int i = 0; i < 9999; i++)
            c.a();
    }
}
class B extends Thread {
    C c;
    public B(C c) {
        this.c = c;
    }
    public void run() {
        for (int i = 0; i < 9999; i++)
            c.b();
    }
}
class C {
    private char a = 'a', b = 'b';
    public synchronized void a() {
        if (a != b)
            System.out.println("a");
        a = 'a';
        b = 'a';
    }
    public void b() {
        if (a != b)
            System.out.println("b");
        a = 'b';
        b = 'b';
    }
    public static void main(String[] args) {
        C c = new C();
        new A(c).start();
        new B(c).start();
    }
}
```

Výstupom tohto programu

- (a) nebude nič alebo budú znaky a a b v nepravidelnom počte a striedaní
- (b) bude desaťtisíkrát znak a
- (c) nebude nič
- (d) budú desaťtisíkrát znaky a a b v nepravidelnom striedaní
- (e) bude desaťtisíkrát znak b

8. (2 b) Čo všetko sa vypíše prostredníctvom príkazov System.out.print() po spustení nasledujúceho programu v Jave (po jeho úspešné alebo neúspešné ukončenie)?

```
class E extends Exception {}

class M {
    public void m(char c) throws E {
        if (c == 'a')
            System.out.print("A");
        else
            throw new E();
    }

    public void f(char c) throws E {
        System.out.print("F");

        try {
            m(c);
        } catch (E e) {
            throw e;
        } finally {
            System.out.print("!");
        }
    }

    public static void main(String[] args) throws E {
        new M().f('a');
        new M().f('b');
        new M().f('a');
    }
}
```

9. (2 b) Potrebné je zabezpečiť rozšíriteľnosť triedy o ďalšie operácie (metódy), ale aby pri pridávaní operácií nebolo potrebné zasahovať do jej kódu. Ktorý návrhový vzor by ste použili?

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

10. (2 b) Hra v Jave obsahuje nasledujúci kód:

```
class Vila {
    private int energia;
    private int zivoty;
    ...
    public void energiaNaZivoty() {
        ziskaneZivoty = energia/100;
        zivoty += ziskaneZivoty;
        energia -= ziskaneZivoty * 100;
        HlavneOkno.okno.pocetZivotov.setText(
            Integer.toString(zivoty));
    }
    ...
}
```

Hlavný problém v tomto kóde z hľadiska objektovo-orientovaného návrhu je to, že

- (a) atribúty sú **private** a odvodeným triedam nebudú prístupné
- (b) kód na prepočítanie energie na životy nie je súčasťou zodpovedajúceho prijímača (spracovateľa)
- (c) vnútorná logika sa mieša s používateľským rozhraním
- (d) metóda mení dva atribúty
- (e) kód na prepočítanie energie na životy nie je súčasťou triedy okna hry

11. (3 b) Čo sa vypíše po spustení nasledujúceho programu v Jave?

```
class A {
    public void x() {
        System.out.print("Ax");
    }
    public static void y() {
        System.out.print("Ay");
    }
}

class B extends A {
    public void x() {
        super.x();
        System.out.print("Bx");
    }
    public static void y() {
        A.y();
        System.out.print("By");
    }
}

class C extends B {
    public void x() {
        System.out.print("Cx");
    }
    public static void y() {
        System.out.print("Cy");
    }
}

class M {
    public static void main(String[] args) {
        B o1 = new C();
        C o2 = new C();
        A o3 = new B();
        B o4 = new B();
        A o5 = new B();

        ((C) o1).x();
        ((C) o1).y();
        System.out.print(" ");

        ((B) o2).x();
        ((B) o2).y();
        System.out.print(" ");

        o3.x();
        o3.y();
        System.out.print(" ");

        o4.x();
        o4.y();
        System.out.print(" ");

        ((A) o5).x();
        ((A) o5).y();
    }
}
```

12. (3 b) Trieda, ktorá reprezentuje špeciálny dokument, je odvodená od triedy, ktorá reprezentuje všeobecný dokument. Metóda na zistenie signatára všeobecného dokumentu nezaručuje vrátenie mena signatára, lebo všeobecný dokument nemusí mať signatára. Táto metóda je pri špeciálnom dokumente prekonaná a zaručuje vrátenie mena signatára zo zoznamu povolených signatárov. Týmto sa predpoklady a dôsledky pôvodnej metódy zoslabujú, zosilňujú alebo sa nemenia? Je týmto dodržaný Liskovej princíp substitúcie (LSP)?

Odpovedzte vo forme: *predpoklady / dôsledky / LSP*. Položky *predpoklady* a *dôsledky* nahraďte jednou z možností *zoslabujú sa*, *zosilňujú sa* alebo *nemenia sa*. Položku *LSP* nahraďte jednou z možností *dodržený* alebo *nedodržený*.

(vypláte tlačným písmom)

Priezvisko:

Meno:

13. (10 b) V urbanistickom simulačnom programe sa vo vzťahu k mestu vyskytujú rôzne dekorácie. Zatiaľ sú to farebné svetlá a fontány, ale v budúcnosti pribudnú ďalšie druhy. V meste sa šetrí energiou a dekorácie obyvateľom naznačujú, kedy je to potrebné. Tak farebné svetlá svietia modro, keď je okamžitá spotreba mesta pod 50% dlhodobého priemeru, žltá, keď je okamžitá spotreba mesta medzi 50% a 70% dlhodobého priemeru, a červená, keď je okamžitá spotreba mesta nad 70% dlhodobého priemeru. Fontány sú v prevádzke len keď je okamžitá spotreba mesta pod 60% dlhodobého priemeru. Implementácia spôsobu počítania dlhodobého priemeru a okamžitej spotreby mesta nie je súčasťou úlohy.

Navrhnite a implementujte v Jave zodpovedajúce objektovo-orientované riešenie zohľadňujúce princípy objektovo-orientovaného programovania. Využite pritom najvhodnejší z návrhových vzorov Strategy, Observer, Visitor a Composite.

Základný návrh predložte vo forme náčrtu diagramu tried v UML, ktorý bude obsahovať najvýznamnejšie vzťahy, operácie a atribúty. Zoberte pritom do úvahy návrhový vzor. Viditeľnosť nie je potrebné uvádzať.

V implementácii sa sústreďte na aplikačnú logiku – GUI nie je predmetom otázky. Taktiež, použité algoritmy nemusia byť optimálne.

Identifikujte explicitne prvky, ktorými sú modelované a implementované roly aplikovaného návrhového vzoru, a vysvetlite, prečo ste ho aplikovali. Poskytnite príklad použitia, v ktorom vytvoríte príslušné objekty a spustíte ich interakciu.

Odpoveď bude hodnotená podľa nasledujúceho kľúča:

- zabezpečenie základnej funkčnosti – 4 b
- kvalita a flexibilita objektovo-orientovaného návrhu – 6 b

1 b

2 d

3 b

4 e

5 c

6 e

7 a

8 FA!F!

9 a

10 c

11 $C_x C_y$ $C_x A_y B_y$ $A_x B_x A_y$ $A_x B_x A_y B_y$ $A_x B_x A_y$

12 nemenia sa / zosilňujú sa / dodržaný

V poslednej otázke mal byť aplikovaný vzor Observer. Mesto alebo údaje o spotrebe energie ako také sú reprezentované abstraktnou triedou alebo rozhraním a hrajú rolu Subjectu, Od tejto triedy alebo rozhrania je odvodený jediný konkrétny typ mesta. Dekorácie sú reprezentované abstraktnou triedou alebo rozhraním a hrajú rolu Observera. Od tejto triedy alebo rozhrania sú odvodené konkrétne druhy dekorácií. V metóde, ktorá slúži na aktualizáciu okamžitej spotreby, sa aktivuje notifikácia všetkých registrovaných dekorácií, ktoré následne upravujú svoj stav.