

Priezvisko:

Meno:

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	

Test trvá 35 minút. V uzavretých otázkach s ponúknutými odpoveďami je vždy správna iba jedna možnosť. Do tabuľky uveďte písmeno pod ktorým je označená odpoveď, ktorú vyberáte. Hodnotia sa len odpovede v tabuľke. V prípade opravy jasne vyznačte odpoveď, ktorá platí. Každá správna odpoveď má hodnotu vyznačenú v otázke. Nesprávna odpoveď, alebo nejednoznačné vyznačenie má hodnotu 0 bodov. Postup riešenia sa nehodnotí. Akceptovaný bude len odovzdaný celistvý list.

1. (1b) Trieda

- popisuje množinu objektov, ktoré majú rovnaké atribúty a správanie. Stav jednotlivých objektov však nemôže byť odlišný
- popisuje množinu objektov, ktoré majú rovnaké vlastnosti a správanie. Stav jednotlivých objektov však môže byť odlišný
- popisuje množinu objektov, ktoré majú rovnaké vlastnosti a správanie. Stav jednotlivých objektov však nemôže byť odlišný
- popisuje množinu objektov, ktoré nemajú rovnaké vlastnosti a správanie. Stav jednotlivých objektov však môže byť odlišný
- popisuje množinu objektov, ktoré nemajú rovnaké vlastnosti a správanie. Stav jednotlivých objektov však nemôže byť odlišný

2. (1b) Daný je nasledujúci kód v Jave:

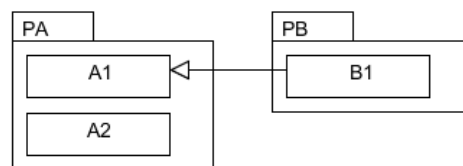
```
public class A {
    void m(int i) { }
    void m(int i, int j) { }
}

public class B extends A {
    void n(int i) { }
    void m(int i, int j) { }
}
```

Ktoré z nasledovného je pravdivé tvrdenie?

- B.n(int i) prekonáva A.m(int i)
- A.m(int i, int j) prekonáva A.m(int i)
- B.m(int i, int j) preťažuje B.n(int i)
- B.n(int i) preťažuje A.m(int i)
- B.m(int i, int j) prekonáva B.n(int i)
- B.m(int i, int j) prekonáva A.m(int i)
- B.m(int i, int j) preťažuje A.m(int i)

3. (1b) Daný je nasledujúci diagram v jazyku UML:



Znáznomený vzťah na úrovni implementácie v jazyku Java znamená, že

- atribúty triedy B1 sú v rámci triedy A1 viditeľné s uvedením modifikátora `public` a `protected`
- atribúty triedy A1 sú v rámci triedy B1 viditeľné s uvedením modifikátora `private` a `protected`
- atribúty triedy B1 sú v rámci triedy A1 viditeľné s uvedením modifikátora `private` a `protected`
- atribúty triedy A1 nie sú v rámci triedy B1 viditeľné s uvedením modifikátora `public` a `protected`
- atribúty triedy B1 sú v rámci triedy A1 viditeľné iba s uvedením modifikátora `public` a `protected`
- atribúty triedy A1 sú v rámci triedy B1 viditeľné s uvedením modifikátora `public` a `protected`

4. (1b) Daný je nasledujúci kód v Jave:

```
public class A {
    static int pocet = 0;
    int p = 1;

    public A(int pocet) {
        this.pocet++;
        this.p = pocet;
        pocet = 3;
    }
}
```

Čo sa vypíše po vykonaní príkazov:

```
new A(2);
System.out.println(A.pocet);
```

- Kompilačná chyba
- 3
- 2
- 1
- 0

5. (1b) Deklarácia `import static java.lang.Math.*;`

- Sprístupní priestor názvov statických atribútov triedy `Math`
- Sprístupní priestor názvov statických metód triedy `Math`
- Sprístupní priestor názvov statických atribútov a metód triedy `Math`
- Naimportuje všetky statické atribúty a metódy triedy `Math`
- Naimportuje všetky statické atribúty triedy `Math`
- Naimportuje všetky statické metódy triedy `Math`

6. (2b) Daný je nasledujúci kód v Jave:

```
public class Prva {
    public Prva() {
        System.out.print("P");
    }
}
public class Druha extends Prva{
    public Druha() {
        System.out.print("D");
    }
}
public class Tretia extends Prva {
    public Tretia() {
        System.out.print("T");
    }
}
```

Čo sa vypíše po vykonaní príkazov:

```
new Druha();
new Tretia();
new Prva();
```

- (a) DPPTP
- (b) PTPDP
- (c) DPTPP
- (d) PDPTP
- (e) DPPPT

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

```
class Clovek {
    String meno, priezvisko;

    Clovek(String meno, String priezvisko) {
        this.meno = meno;
        this.priezvisko = priezvisko;
    }

    public String toString() {
        return meno + " " + priezvisko;
    }

    void vypis() {
        System.out.println(priezvisko + " " + meno);
    }
}

class Student extends Clovek {
    int rocnik;

    Student(String meno, String priezvisko, int rocnik) {
        super(meno, priezvisko);
        this.rocnik = rocnik;
    }

    public String toString() {
        return priezvisko + " " + meno;
    }

    void vypis() {
        System.out.println(meno + " " + priezvisko);
    }
}
```

Ktorý z príkazov zabezpečí výpis na konzolu v tvare:

Adam Prvy

- (a) System.out.println(new Clovek("Adam", "Prvy"));
- (b) System.out.println(new Student("Adam", "Prvy", 1));
- (c) new Clovek("Adam", "Prvy").vypis();
- (d) System.out.println(new Student("Adam", "Prvy", 1).vypis());
- (e) new Student("Adam", "Prvy", 1).super.vypis();

8. (1b) Inštancia triedy

- (a) má svoj stav a správanie ale nemá identitu
- (b) nemá stav, ale má správanie a identitu
- (c) má identitu, stav ale ešte nemá správanie
- (d) nemá identitu ani stav ale má správanie
- (e) má svoj stav, správanie a identitu

9. (1b) Dedenie nám v odvodenej triede neumožňuje:

- (a) meniť štruktúru rodičovskej triedy
- (b) ponechať a využívať všetko z rodičovskej triedy
- (c) doplniť, čo v rodičovskej triede nebolo a v odvodenej chýba
- (d) zmeniť to, čo z rodičovskej triedy nevyhovuje
- (e) preťažovať metódy rodičovskej triedy

10. (1b) Stav objektu triedy

- (a) je definovaný bezparametrickým konštruktorom
- (b) je definovaný inštanciou agregovanej metódy
- (c) je definovaný hodnotami jeho atribútov
- (d) je definovaný modifikátormi prístupu
- (e) je nedefinovaný

11. (1b) Medzi triedou A a triedou B je vzťah agregácie.

Tento vzťah:

- (a) reprezentuje dedenie vlastností nadtypu
- (b) predstavuje princíp znovupoužitia programového kódu skladaním
- (c) znamená dedenie vlastností podtypu
- (d) umožňuje skrývanie implementácie objektu
- (e) poskytuje možnosť meniť vlastnosti oboch tried

12. (1b) Deklarácia nestatickej metódy rovnakej signatúry v Podtype

- (a) preťažuje (overloads) pôvodnú metódu Nadtypu
- (b) umožňuje jej dedenie tým, že je nestatická
- (c) neumožňuje jej dedenie práve preto, že je nestatická
- (d) preťažuje (overloads) pôvodnú metódu Podtypu
- (e) prekonáva (overrides) pôvodnú metódu Nadtypu

13. (1b) Rozhranie poskytované objektom

- (a) reprezentuje stav inštancie
- (b) je súborom jeho atribútov
- (c) vymedzuje požiadavky, ktoré naň môžu byť kladené
- (d) popisuje množinu inštancií triedy, ktoré majú rovnaké vlastnosti a správanie
- (e) popisuje množinu inštancií triedy, ktoré nemajú rovnaké vlastnosti a správanie

Spolu 15 bodov

Riešenie:

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