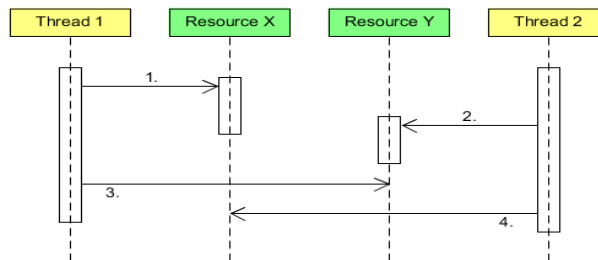


Szálkezelés

1. A szekvencia diagram feladata az objektumok egymás közti üzenetváltásainak ábrázolása egy időtengely mentén elhelyezve. Az objektumok életvonala egy felülről lefelé mutató időtengely. A nyilak az objektumok egymásnak szóló üzenetei.

Az alábbi ábrán a nyilak az erőforrások lock-olását jelölik. A kezdetektől tudjuk, hogy mindkét szálnak szüksége lesz mindkét erőforrásra.



Melyik az a hívás, amelynek megtörténtekor már biztosak lehetünk a deadlock kialakulásában?

2. Mi lesz az eredménye egy olyan objektum `wait()` metódusának hívásának, amelynek a hívó szál nem birtokolja a monitor lockját?

3. Mi a szálak alapértelmezett prioritása?

4. Melyik output nem lehetséges a következő kód futtatása esetén?

```
public class Tester extends Thread {
    int code = 9;
    public void run() {
        this.code = 7;
    }
    public static void main(String[] args) {
        Tester thread = new Tester();
        thread.start();
        for (int i = 0; i < 5; i++) {
            System.out.print(thread.code);
        }
    }
}
```

A – 97777 B – 77777 C – 79999 D – 99999

5. Mi lesz a következő kód futtatásának eredménye?

```
public class MyThread extends Thread {
    volatile private int x = 3;
    public static void main(String[] args) throws Exception {
        new MyThread().foo();
    }
    public MyThread() {
        x = 10;
        start();
    }
    public void foo() throws Exception {
        join();
        x = x - 1;
        System.out.println(x);
    }
    public void run() { x *= 2; }
}
```

A – 8

B – 19

C – 18

D – 10

6. Mi lesz a következő kód futtatásának eredménye?

```
public class Test extends Thread {  
    public void run() {  
        System.out.println(isAlive());  
    }  
    public static void main(String[] args) {  
        Test test = new Test();  
        System.out.println(test.isAlive());  
        test.run();  
    }  
}
```

A – false false

B – false true

C – true false

D – true true

7. Melyek Atomi műveletek az alábbiak közül?

- A. Referencia változók írása/olvasása
- B. Minden primitív típusú változó írása/olvasása
- C. Mindkettő
- D. Egyik sem

8. Melyik állítás igaz az alábbi kódra?

```
public class Test extends Thread {  
    private static int x;  
    public synchronized void foo() {  
        int current = x;  
        current++;  
        x = current;  
    }  
    public void run() {    foo();    }  
}
```

- A. A run metódus szinkronizálása szálbiztossá tenné az osztályt
- B. Az osztály szálbiztos
- C. A foo metódus statikussá tétele szálbiztossá tenné az osztályt
- D. Futtatáskor exception – t eredményez

9. Melyik állítás nem igaz az immutable osztályokra?

- A. Az osztályt final – ként kell deklarálni a leszármaztatás megelőzése végett
- B. Az osztály összes adattagja private final – ként kell deklarálni
- C. Az objektum típusú adattagokról mindig másolatot kell készíteni amikor azt getter metódusból ki kell adni és konstruktor paraméterből való inicializáció
- D. Az osztály nem tartalmazhat olyan metódusokat, amelyek az állapotát a konstruktor lefutása után megváltoztatják.

10. Melyik állítások igazak a Concurrent API – val kapcsolatban?

- 1. Az ExecutorService execute metódusa Runnable és Callable objektumokat is elfogad.
- 2. Az ExecutorService submit metódusa kizárólag Callable objektumot fogad el és egy Future objektummal tér vissza.
- 3. A Future objektum fogja tartalmazni a Callable objektumként definiált szál futásának az eredményét.
- 4. A Lock típusú objektumot – hasonlóan a szinkronizációkor használt monitor lock-hoz – egy időben csak egy szál birtokolhatja.

Adatbáziskezelés / JDBC

1. Melyik nem DML utasítás az alábbi SQL utasítások közül?

- A. update [table] set ...
- B. delete from [table] ...
- C. alter table ...
- D. insert into [table] ...

2. Mely állítások igazak a JDBC kapcsolat felépítésével kapcsolatban?

- 1. DriverManager osztályon keresztül juthatunk pool-ozott kapcsolathoz.
- 2. DataSource osztályon keresztül juthatunk pool-ozott kapcsolathoz.
- 3. DataSource osztályok történő kapcsolódás esetén meg kell adnunk az adatbázishoz tartozó connection URL-t.
- 4. Adatbázis kapcsolatot a DriverManager.getConnection hívással kaphatunk.

A – 1, 3

B – 2, 3

C – 2, 4

D – 3, 4

3. Mely információ nem nyerhető ki az SQLException objektumból?

- A. SQL státusz kód.
- B. Driver/Adatbázis specifikus hibakód.
- C. A hibát okozó adatbázis kérés.
- D. A felmerült hiba leírása.

4. Melyik helytelen módja a resultSet adatainak elérésének?

- A. String value0 = rs.getString(0);
- B. String value1 = rs.getString(1);
- C. int value2 = rs.getInt(2);
- D. int value3 = rs.getInt("ADDR_LN1");

5. Melyik állítás igaz az alábbiak közül?

- 1. CallableStatement kiterjeszti a PreparedStatement interface-t. Ez az interface használható SQL tárolt eljárások hívására.
- 2. Statement kiterjeszti a PreparedStatement interface-t és akkor használatos, amikor az SQL lekérdezést nem szükséges többször futtatnunk.
- 3. PreparedStatement statikus lekérdezések indítására használatos (pl.: select * from table), ezért PreparedStatement-ek nem paraméterezhetők.
- 4. PreparedStatement használatával lehetséges SQL utasítások batch feldolgozása.

6. Hogyan indítható új adatbázis tranzakció?

- A. A Connection-höz egy Transaction object kérésével és azon begin() metódus hívással.
- B. A Connection-höz egy Transaction object kérésével és annak autoCommit tulajdonságának false-ra állításával.
- C. A Connection beginTransaction metódusának hívásával.
- D. A Connection autoCommit tulajdonságának false-ra állításával és egy SQL utasítás végrehatásával.

Szoftvertechnológia

1. Melyek a szoftverek alapvető minőségi mutatói?

- A. Szállítási idő, megvalósítási költség, hardver- és szoftverigény.
- B. Karbantarthatóság, megbízhatóság, biztonság, hatékonyság, használhatóság.
- C. Módosíthatóság, bővíthetőség, felbonthatóság, újrahasználhatóság, megbízhatóság.
- D. Ergonómia, használhatóság, kompatibilitás, hardver- és szoftverigény.

2. Melyik a használati történet (user story) szerkezete?

- A. USER felhasználó IN USE CASE használati eset WITH RELATION kapcsolat
- B. AS A szerepkör USE funkció TO cél
- C. WHEN tevékenység APPLYING funkció IN ORDER TO cél
- D. GIVEN környezet WHEN tevékenység THEN hatás

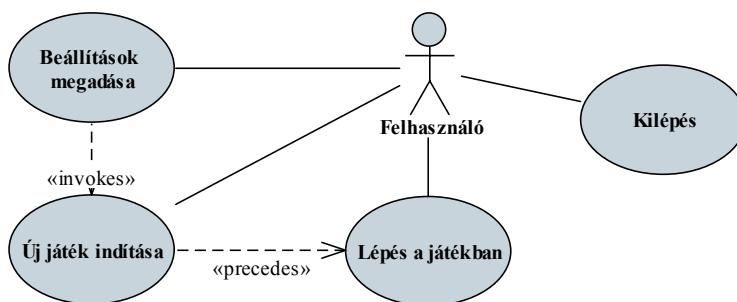
3. Mi a helyes sorrendje a követelményelemzésnek?

- A. megvalósíthatósági elemzés, követelmény feltárás, követelmény specifikáció, követelmény validáció
- B. követelmény feltárás, követelmény specifikáció, követelmény validáció, megvalósíthatósági elemzés
- C. követelmény feltárás, követelmény validáció, követelmény specifikáció, megvalósíthatósági elemzés
- D. követelmény feltárás, követelmény specifikáció, megvalósíthatósági elemzés, követelmény validáció

4. Az alábbiak közül melyik nem szoftverfejlesztési modell?

- A. waterfall
- B. evolution
- C. scrum
- D. prototyping

5. Melyik funkcionalitás olvasható ki az alábbi használati esetek diagramból?



- A. A felhasználónak lehetősége van új játékot kezdeni, de csak miután a beállításokat megadta.
- B. A felhasználó a beállítások megadásával automatikusan új játékot indít.

- C. A felhasználónak külön nem szükséges beállításokat megadni, vagy új játékot indítania, azonnal léphet a játékban.
- D. A felhasználó csak akkor léphet ki a programból, ha elkezdett egy játékot.

6. Az alábbiak közül melyek az UML használati eset (use case) diagram relációi?

- A. függőség (dependency), kompozíció (composition), használat (usage), beágyazás (nesting)
- B. előfeltétel (precedes), tartalmazás (include), használat (usage), általánosítás (generalization)
- C. használat (usage), beágyazás (nesting), importálás (import), függőség (dependency)
- D. felület (interface), megvalósítás (implementation), tartalmazás (include)

7. Mire szolgál az UML telepítési (deployment) diagram?

- A. Ábrázolja azt a műveletsorozatot, amely a szoftver adott gépen történő üzembehelyezéséhez szükséges.
- B. Ábrázolja az összes hibalehetőséget, amellyel a telepítés során találkozhatunk.
- C. Ábrázolja a szoftver összetevőket, annak megfelelően, miként kell őket telepítőcsomagba helyezni.
- D. Ábrázolja a szoftver összetevőinek fizikai (különböző gépeken történő) elhelyezését, a szükséges szoftverkörnyezettel.

8. Mely objektumorientált elvet sérti az egyke (singleton) szerkezet, amely korábban egy népszerű tervminta volt? Az egyke szerkezet azt garantálja, hogy az objektumból csak egy példány legyen, amelyet egy statikus művelet segítségével kérhetünk el az osztálytól.

Singleton	
-	<u>instance</u>
+	<u>Instance()</u> if (instance == null) instance = new Singleton() return uniqueInstance
-	Singleton()

- A. Single Responsibility Principle
- B. Open/Closed Principle
- C. Liskov Substitution Principle
- D. Dependency Inversion Principle

9. Az alábbiak közül melyik technika használható a Dependency Inversion Principle megvalósítására?

- A. (figyelő) observer
- B. MVC (modell-view-controller)
- C. függőség befecskendezés (dependency injection)
- D. általánosítás (generalization)

10. Mit jelent a tesztvezérelt fejlesztés (TDD)?

- A. Szoftverfejlesztési módszer, amelyben a tesztek a tényleges programkód elkészítése előtt írják meg.
- B. Tesztelési módszer, amelynek célja, hogy az tesztesetek minden progamegységre kiterjedjenek, és megfelelő sorrendben hajtódnak végre.
- C. Egy általános elv, amely kimondja, hogy a programkód minden utasítását ellenőrizni kell egységtesztek segítségével (100%-os kódlefedettség).
- D. Tesztelési módszer, amelyben először egységteszteket készítenek az osztályokra (és metódusaikra), majd integrációs tesztekkel ellenőrzik az osztályok együttes viselkedését, végül rendszertesztekkel a teljes szoftvert viselkedését ellenőrzik.

11. Az alábbiak közül mely funkciót nem tudják biztosítani a teszt keretrendszerek (unit testing frameworks)?

- A. Tesztesetek manuális létrehozását külön progamegységekben (osztályokban).
- B. Mindent lefedő tesztesetek automatikus generálását a programkód elemzésével.
- C. A kapott és elvárt eredmények összehasonlítását elvégző assert utasításokat.
- D. Tesztjelentés elkészítését, amelyben látható, hogy mely tesztek lettek sikeresek/sikertelenek.

További kérdések

1. Mit kell implementálni saját típus esetén a HashMap használatához?

- A. == operátor
- B. hashCode(...) metódus
- C. equals(...) metódus
- D. Mindkét metódust

2. Miből indul ki az objektum orientált tervezés?

- A. Funkciók
- B. Tevékenységek
- C. Entitások és kapcsolataik
- D. Architektúra

3. Mivel fejezzük ki, hogy egy objektum egy osztály több objektumával is kapcsolatban áll?

- A. Kompozícióval
- B. Multiplicitással
- C. Tömb típusú attribútummal
- D. Aggregációval

4. Mihez rendelhetünk egy osztálydiagram esetén megszorítást?

- A. Reláció
- B. Attribútum
- C. Metódus paraméterek
- D. Mindhez

5. Mely reláció hat osztályok között?

- A. Asszociáció
- B. Függőség
- C. Származtatás
- D. Aggregáció

6. Mit támogat a Java a többszörös specializáció és többszörös általánosítás közül?

- A. Általánosítás
- B. Specializáció
- C. Mindkettő
- D. Egyik sem

7. Mely állítás igaz?

- A. final abstract osztálynak legalább egy metódusa abstract
- B. abstract osztálynak legalább egy metódusa abstract
- C. final osztály minden attribútuma final
- D. abstract osztály leszármazottja is lehet abstract

8. Mely állítás nem igaz?

- A. final osztály nem származtatható
- B. interface-ek nem származtathatók egymásból
- C. Egy osztály több interface-t is megvalósíthat
- D. Az interface minden metódusát meg kell valósítani

9. Mi lehet generikus paraméter?

- A. Alaptípus
- B. Osztály
- C. interface
- D. Osztály, ami megvalósítja a generikusban használt műveleteket

10. Mely gyűjtemény indexelhető?

- A. HashSet
- B. HashMap
- C. Vector
- D. TreeMap

11. Mi lehet statikus?

- A. Adattag
- B. Metódus
- C. Osztály / interface
- D. Felsorolási típus

12. Mire való a tranzakció az adatbázisoknál?

- A. Több művelet atomikénti végrehajtására
- B. Átutalások elnevezése
- C. Kapcsolt táblás lekérdezésre
- D. Tárolt eljárások futtatására

13. Mi nem része az Egyed-Kapcsolat diagramnak?

- A. Attribútum
- B. Entitás
- C. Kulcsok
- D. Osztály

14. Mit szokás kapcsoló táblával leképezni?

- A. Kapcsolt táblás lekérdezést
- B. 1 – n kapcsolatot
- C. 1 – 1 kapcsolatot
- D. m – n kapcsolatot

15. Melyik nem igaz a JTable es a modell kapcsolatáról?

- A. A JTable nem tartalmaz adatot
- B. A modell felüldefiniálható
- C. A modell nem tudja értesíteni a JTable-t a változásról
- D. A modell reprezentációja eltérhet a JTable által lekérdezett adatokétól

16. Mi nem része a dinamikus modellnek?

- A. Állapot diagram
- B. Szekvencia diagram
- C. Tevékenység diagram
- D. Komponens diagram

17. Mije nem lehet az állapotnak?

- A. Neve
- B. Invariánsa
- C. Előfeltétele
- D. Paramétere

18. Mivel csökkenthető az állapotdiagram komplexitása?

- A. Általánosítás
- B. Aggregáció
- C. Általánosítással és aggregációval
- D. Más módszerrel

19. Melyik igaz?

- A. Az általánosítás invariánsa az állapotok invariánsainak diszjunkciója
- B. Az általánosítás invariánsa az állapotok invariánsainak konjunkciója
- C. Az aggregáció invariánsa az állapotok invariánsainak diszjunkciója
- D. A két módszerrel kapott állapotoknak nincs invariánsa

Scrum

1. Melyik nem agilis elv a következők közül?

- A. a módszertan érvényesítése, szemben az eszközökkel
- B. a működő szoftver, szemben az átfogó dokumentációval
- C. együttműködés az ügyféllel, szemben a szerződéses tárgyalásokkal
- D. a változásra való reagálás, szemben a terv követésével.

2. Melyik állítás nem igaz a futam kapcsán?

- A. A terméknek mind a tervezése, kódolása és tesztelése is a futamon belül történik.
- B. A futam eredménye üzleti értéket képviselő működő kód.
- C. A feladatok és az idők meghatározása után csak a termékgazda szól bele a csapat munkájába.
- D. A Scrum csapat önszerveződő módon dolgozik a futam során.

3. Melyik állítás igaz a Scrum mesterre?

- A. A Scrum mester a Scrum csapat menedzsere
- B. A Scrum mester vezeti a napi Scrumot
- C. A Scrum mester nem felel azért, hogy külső hatásoktól védje a SCRUM csapat munkáját
- D. A Scrum mester a folyamatokért felel

4. Melyik állítás igaz a napi Scrum-ra?

- A. A napi Scrum-ot a Scrum mester vezeti.
- B. A napi Scrum során a csapattagok beszámolnak a Scrum mesternek a haladásukról.
- C. A napi Scrum során az a cél, hogy felszámoljuk a csapatot érintő akadályokat.
- D. A napi Scrum maximum 15 percet tarthat.

5. A három Scrum termék a következő:

- A. termék kívánságlista, futam feladatlista, Scrum tábla
- B. termékvízió, termék kívánságlista, felhasználói történet
- C. termék kívánságlista, Scrum tábla, haladási diagram
- D. termék kívánságlista, futam feladatlista, inkrementum