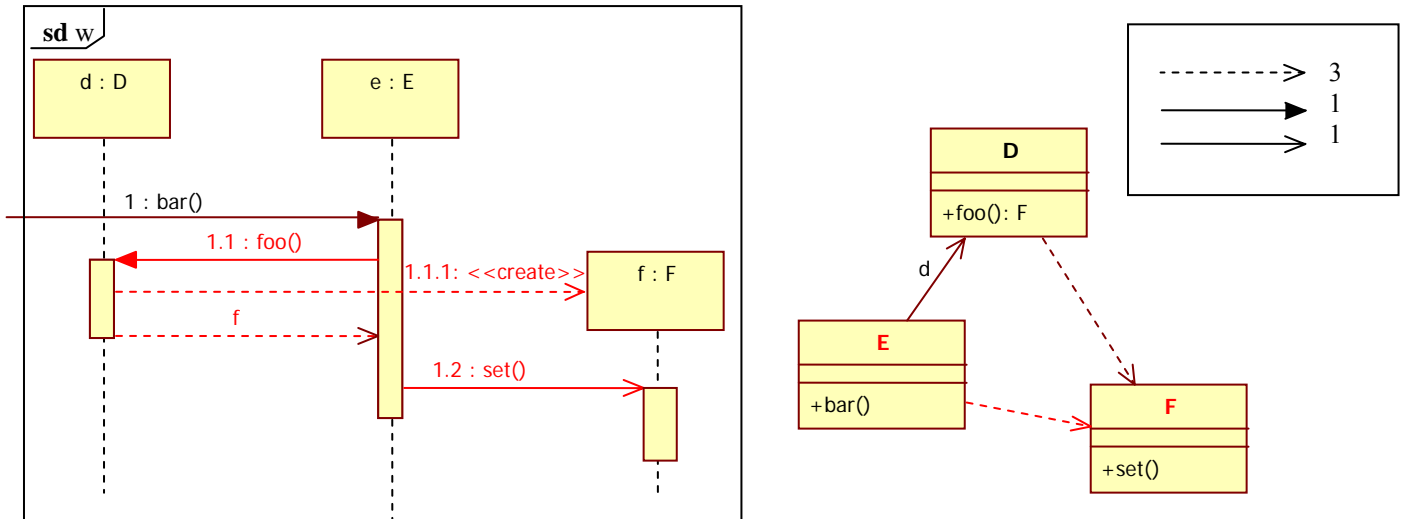


VIZSGA FELADATSOR SZOFTVERTECHNOLÓGIA

c. tárgyból
2012. január 3.

Az első lapon található feladatok megoldására 30 perc áll rendelkezésére. Az elérhető 24 pontból minimum 14 pontot kell kapnia ahhoz, hogy a második lapon szereplő feladatokra adott megoldásait értékeljük.

1. Az ábrán egy UML2 osztálydiagram és egy UML2 szekvenciadiagram látszik. A két diagram szemantikailag összefügg, de hiányos. Rajzolja be a hiányzó jelölő-elemeket! Ahol lehet, lássa el őket feliratokkal is! A felhasználható jelölőelemek és számosságuk a mellékelt keretben látható. (5 pont)



2. A kulcs felhasználásával jellemezze a Java nyelvvel kapcsolatos állításokat ! (3 pont)

- | | |
|--|--------|
| A - csak az első tagmondat igaz | (+ -) |
| B - csak a második tagmondat igaz | (- +) |
| C - mindkét tagmondat igaz, de a következtetés hamis | (++ -) |
| D - mindkét tagmondat igaz és a következtetés is helyes | (++ +) |
| E - egyik tagmondat sem igaz | (- -) |

- [E]** Egy *X* osztály *foo* synchronized metódusa nem lehet final, mert az *X* leszármazottaiban a *foo* metódusban az őosztály egy objektumának monitorát kellene használni.
- [B]** A private hozzáférési osztályú attribútumok nem szerializálódnak, mert ezeket csak az adott osztály metódusai érhetik el közvetlenül.
- [A]** A *java.awt.event* csomagban a *WindowAdapter* osztály megvalósítja a *WindowListener* interfészt, mert a *WindowAdapter* osztály használatával lehetővé válik a többszörös öröklés.

Blank 0 pont, minden találat 1 pont, minden rossz válasz -0.5 pont, de total >= 0

Mit ír ki az alábbi Java kódrészlet? Válaszát írja a négyzetrácsba! (1 pont)

```
Map<String, Integer> ht = new HashMap<String, Integer>();
for (int i = 0; i < 3; i++) {
    if (!ht.containsKey("alma"))
        ht.put("alma", 1);
    Integer n = ht.get("alma");
    n += i;
}
System.out.println(ht.get("alma"));
```

1							

3. Jelölje (karikázza be) az állítások igazságtartalmát, ha feltesszük, hogy szabványos Java nyelvet használunk! (4 pont)

- I H** Egy objektum referenciáját tartalmazó változón csak olyan metódus hívható meg, amilyen a változó statikus típusában is szerepel.
- I H** Egy változó dinamikus típusa nem lehet absztrakt osztály.
- I H** Előfordulhat, hogy két szál (T1 és T2) ugyanazon objektum ugyanazon synchronized metódusát futtatva T1 T2 sorrendben lép be, de T2 T1 sorrendben lép ki.
- I H** Ha a t tömböt paraméterként átadjuk egy metódusnak, amelyik a tömb egy elemének új értéket ad, akkor az eredeti tömbben is módosul az érték.

Blank 0 pont, minden találat 1 pont, minden rossz válasz -1 pont, de total >= 0

4. Az alábbi táblázatban a fontosabb szoftver architektúra típusokat adtuk meg, minden típushoz egy indexet rendelve. A jobboldali táblázatban szoftver architektúrákkal kapcsolatos fogalmak állnak. Adja meg, hogy az egyes fogalmak melyik architektúra típushoz kapcsolhatóak leginkább ! (A fogalom előtti mezőbe írja be a típus indexét !) Ha a fogalom egyik típushoz sem társítható, akkor a mezőbe tegyen X-et ! (3 pont)

1	esemény alapú implicit hívást használó (event-based implicit invocation)
2	csővezeték+szűrők (pipes and filters)
3	tábla (blackboard)
4	interpreter
5	szolgáltatás orientált architektúra (SOA)

2	batch (kötegelés)
4	virtuális gép (virtual machine)
1	regisztráció (registration)
3	központi adatbázis (repository)
5	kötés (binding)
1	szórás (broadcast)

5. Jelölje be az alábbi táblán, hogy az egyes szerződéses feltételek megszegése esetén melyik fél a hibás! (3 pont)

	kliens	szerver
előfeltétel (precondition)	X	
invariáns (invariant)		X
utófeltétel (postcondition)		X

Blank 0 pont, minden találat 1 pont, minden rossz válasz -1 pont, de total >= 0

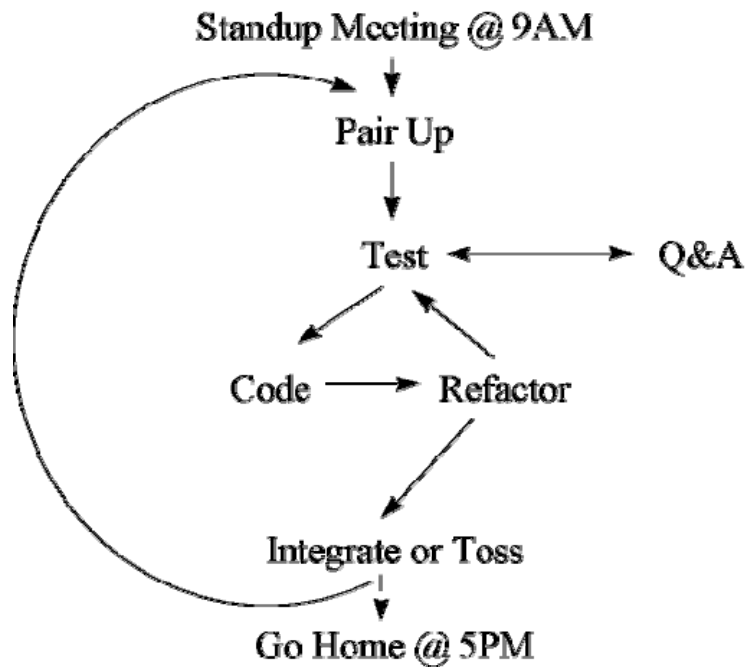
6. Készítse el azt az állapottáblát, amely megfelel az alábbi DTD-vel definiált adatszerkezetnek ! A tábla az 1. állapotban kezdődik, és segítségül beírtuk az a illetve b esetén következő állapotokat. Kötőjellel (-) jelölje, ha egy elem, egy állapotban nem következhet! (5 pont)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE x [
  <!ELEMENT x ((a|(b+,c?)),c)*>
  <!ELEMENT a (#PCDATA)>
  <!ELEMENT b (#PCDATA)>
  <!ELEMENT c (#PCDATA)>
]>
```

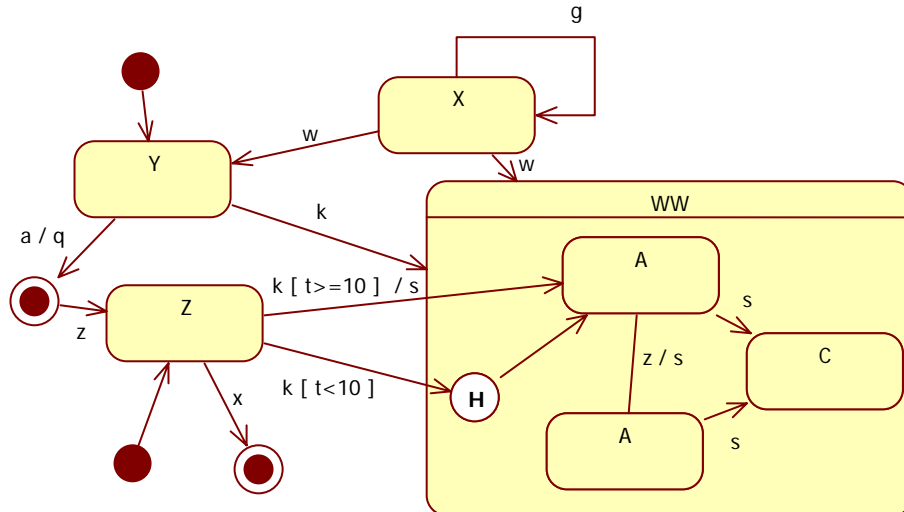
	a	b	c
1	2	3	-
2	-	-	1
3	-	3	4
4	2	3	1

A következő feladatokat csak akkor értékeljük, ha az előző lapon szereplő feladatokból minimum 14 pontot ért el.

7. Írja le (folyamatábrával) az XP módszertan szerinti fejlesztésben egy nap történéseit! (6 pont)



8. Milyen szintaktikai és szemantikai hibák találhatók az alábbi UML2 állapot-diagramon (state-chart)? (8 pont)

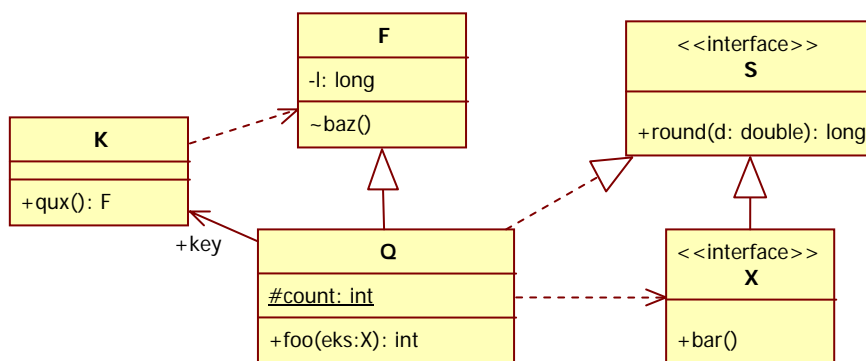


- Két kezdőállapot
- Ugyanarra az eseményre két állapotba is mehet
- Végállapotból kilépünk
- Nincs kezdőállapot
- Ugyanolyan nevű állapotok
- Nincs nyíl
- Csak kimenő állapot
- Csak bemenő állapot

9. Az alábbi Java kódrészletek alapján rajzoljon UML2 osztálydiagramot! (8 pont)

```

class Q implements S extends F {
    protected static int count;
    public K key;
    public int foo(X eks) {
        eks.bar();
    }
}
interface X extends S {
}
interface S {
    long round(double d);
}
class F {
    private long l;
    long baz() { return l; }
}
class K {
    F qux() { return new F(); }
}
    
```



10. A mellékelt táblázatban jelölje be, hogy a felsorolt fogalmak az UML2 4-rétegű meta-modell szerkezetének melyik rétegébe tartoznak! (4 pont)

	M0	M1	M2	M3
Izidor	X			
Actor			X	
State			X	
Autó		X		
Barnabás	X			
UseCase			X	
Ügyfél		X		
Ember		X		

Eredmények értékelése:

Pontszám	Osztályzat
21 -	2
28 -	3
35 -	4
42 -	5