

VIZSGA FELADATSOR

SZOFTVERTECHNOLÓGIA

c. tárgyból

2015. január 20.

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.

A tesztkérdésekre adott rossz válasz esetében pontot veszít, de feladatonként a total pontszám ≥ 0

A feladatsor mentes az aktuálpolitikai témákra való utalásoktól és nem célja, hogy az államigazgatás bármely szervébe vetett közbizalmat megingassa, a tisztviselők iránti közmegebecsülést hátrányosan befolyásolja. A feladatsor szereplői nem valós személyek és bármilyen hasonlóság a valósággal csak a véletlen műve.

1. Jelölje az állítások igazságtartalmát, ha feltesszük, hogy szabványos Java nyelvet használunk! (8 pont)

igen	nem	
<input type="checkbox"/>	<input type="checkbox"/>	catch blokkból szabad exception-t dobni
<input type="checkbox"/>	<input type="checkbox"/>	<i>InputStreamReader</i> konstruktora kaphatja paraméterül a <i>System.in</i> -t
<input type="checkbox"/>	<input type="checkbox"/>	<i>JFrame</i> layoutmanager-e csak a <i>BorderLayout</i> lehet
<input type="checkbox"/>	<input type="checkbox"/>	ha egy <i>s</i> <i>String</i> objektumon meghívjuk a <i>toUpperCase()</i> metódust, akkor módosul az <i>s</i> objektum tartalma
<input type="checkbox"/>	<input type="checkbox"/>	<i>final</i> módosítójú osztálynak nem lehet <i>abstract</i> metódusa
<input type="checkbox"/>	<input type="checkbox"/>	<i>final</i> metódus nem módosíthatja az attribútumok értékét
<input type="checkbox"/>	<input type="checkbox"/>	<i>synchronized</i> módosítójú metódusban nem lehet <i>wait()</i> metódust hívni
<input type="checkbox"/>	<input type="checkbox"/>	<i>foreach</i> ciklusban (pl. <code>for (A a : x){ ... }</code>) nem szabad az iterált kollekciónak (a példában <i>x</i>) új elemet hozzáadni.

2. 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 egyértelműen, akkor a mezőbe tegyen X-et! (3 pont)

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

3	grafikus interfész (GUI)
5	regisztráció (registration)
2	kötés (binding)
X	egységbezárás (encapsulation)
6	pszeudokód (pseudocode)
2	koreográfia (choreography)

3. Subversion-ben update-kor ki változhat? (1 pont)

Igen Nem

munkapéldány

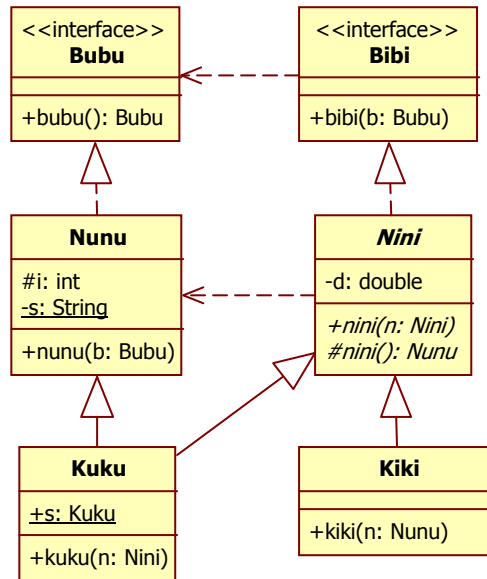
Igen Nem

repository

Az alábbi táblázatba írja be, hogy a munkapéldány és a repository állapota függvényében mi történik update esetén. X-szel jelölje, hanem történik semmi! (4 pont)

	out of date	current
unchanged	reread	X
changed	merge	X

4. Az alábbi UML2 diagram alapján - a kulcs felhasználásával - jellemezze az állításokat ! (8 pont)

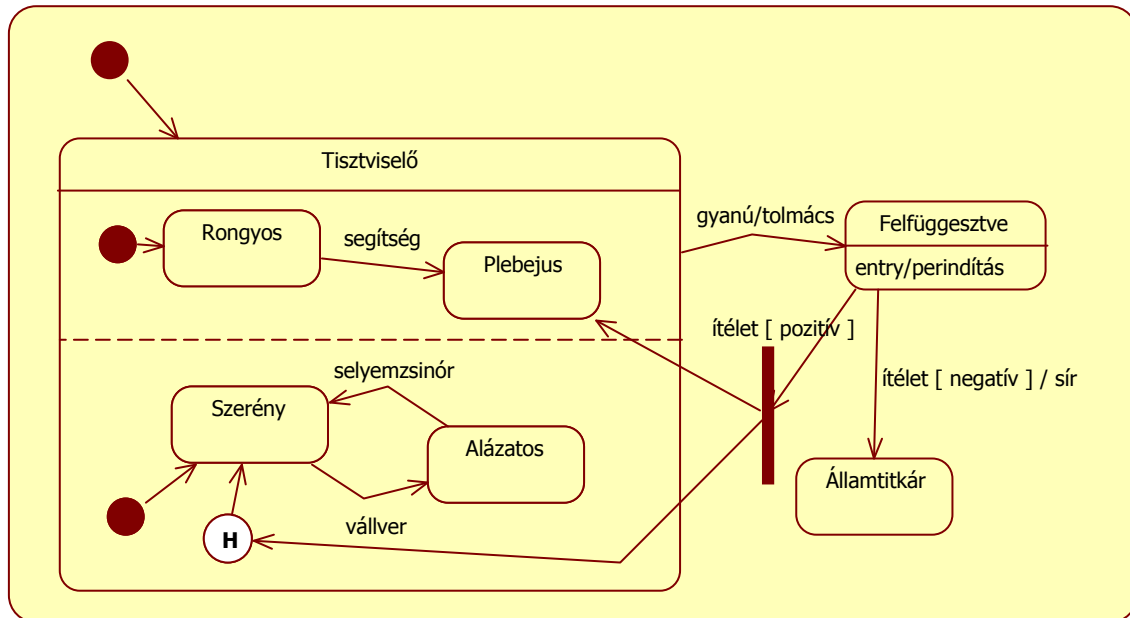


- A - csak az első tagmondat igaz (+ -)
 B - csak a második tagmondat igaz (- +)
 C - mindkét tagmondat igaz, de a következtetés hamis (+ + -)
 D - mindkét tagmondat igaz és a következtetés is helyes (+ + +)
 E - egyik tagmondat sem igaz (- -)

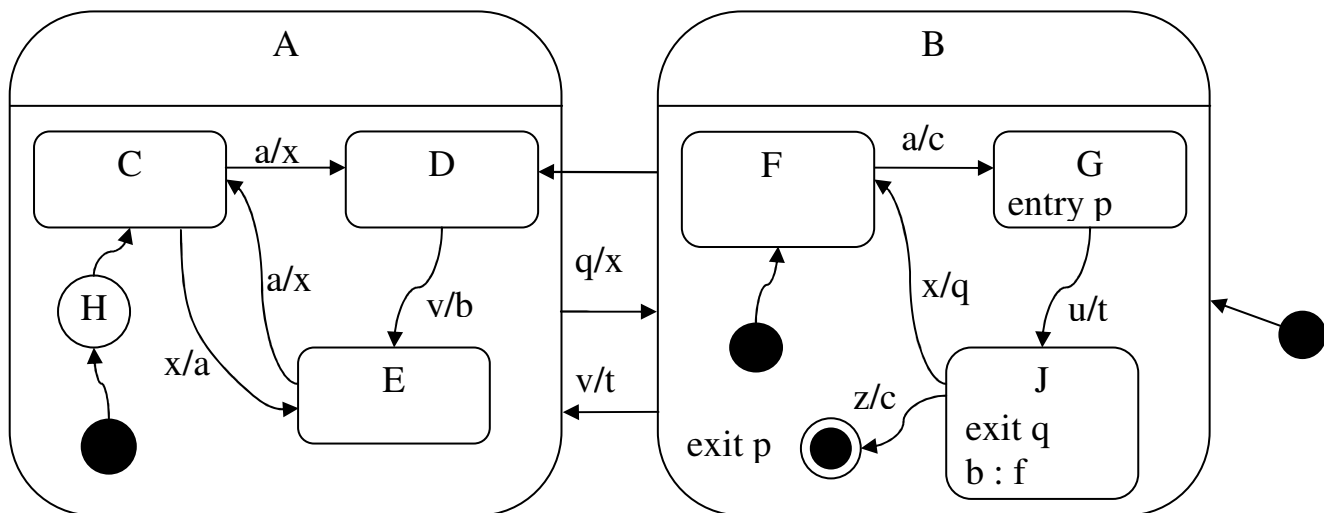
- [B] Nini bibi függvénye nem kaphat paraméterül Kuku objektumot, mert Bibi nem ismeri Kuku-t.
- [C] Van olyan s attribútum, amelyet nem módosíthat Kiki kiki függvénye, mert nem minden s attribútum publikus.
- [C] Kuku implementálja a Bubu és Bibi interfészeket, mert Java-ban megengedett az interfészek többszörös öröklése.
- [C] Nunu nunu függvénye nem példányosíthat Kiki objektumot, mert Kiki függ Nunu-tól.
- [E] Kuku nem példányosítható, mert nem implementálja az absztrakt nini függvényeket.
- [B] Kuku és Kiki interfésze megegyezik, mert mindkettő megvalósítja a Bibi interfészt.
- [E] Kiki kiki függvénye nem kaphat paraméterül Kuku objektumot, mert ha kapna, az sértené a Liskov-elvet.
- [E] Kuku bubu függvénye nem módosíthatja az i attribútum értékét, mert i statikus.

5. Készítsen UML2 állapotdiagramot (statechart) az alábbi történet alapján! (8 pont)

Lilliputban a tisztségviselők rongyosan kezdik tevékenységüket, ekkor még szerények is. Ha családi segítséget kapnak, akkor plebejussá válnak. A szerény tisztségviselő, ha vállon veregetik, alázatos lesz. Ha alázatos, és selyemzsinórt kap, újra szerénnyé válik. Ha a tisztségviselőt gyanú éri, akkor tolmácsot kér és felfüggesztett állapotba kerül. Felfüggesztéskor pert indít. A bírósági ítélet alapján (pozitív esetben) újra tisztségviselő lesz (mindig plebejus, valamint vagy szerény, vagy alázatos, attól függően, hogy a felfüggesztés előtt mi volt). Ha a bírósági ítélet negatív, akkor elsírja magát, és államtitkárrá léptetik elő.



6. A következő UML2 állapotdiagram alapján minősítse az állításokat! (6 pont)



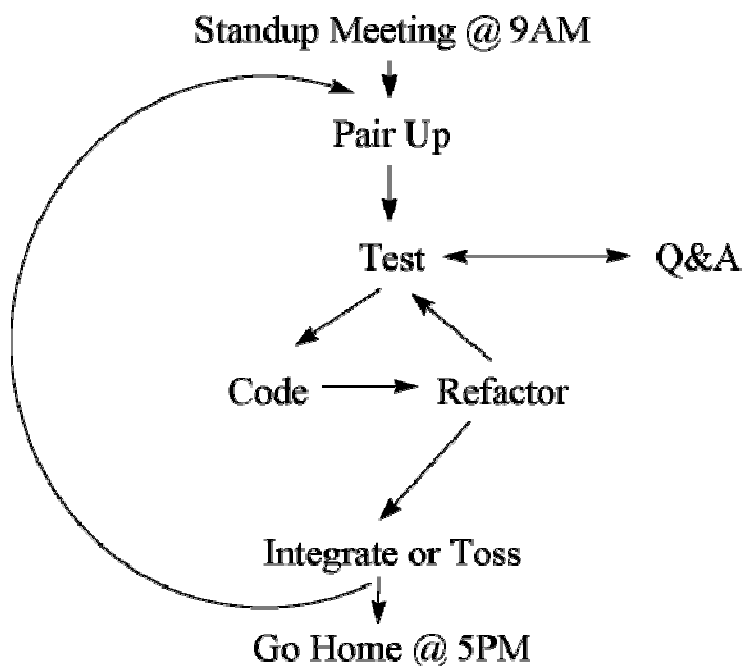
A kezdés után az **a, u, b, z, a, v** esemény-szekvencia hatására

Igaz	Hamis	Állítás
<input type="checkbox"/>	<input type="checkbox"/>	összesen 5 különböző állapotot érintünk
<input type="checkbox"/>	<input type="checkbox"/>	„b” tevékenységet nem végzünk
<input type="checkbox"/>	<input type="checkbox"/>	C lesz a végállapot
<input type="checkbox"/>	<input type="checkbox"/>	van olyan esemény, amelynek hatására nem végzünk tevékenységet
<input type="checkbox"/>	<input type="checkbox"/>	az E állapotot kétszer érintjük
<input type="checkbox"/>	<input type="checkbox"/>	egyszer lefut az „a” tevékenység

7. Mi történik refaktoráláskor (refactoring) az XP agilis módszertanban ? (2 pont)

A szoftvert úgy fejlesztjük tovább, hogy a külső viselkedés változatlan marad, de a belső szerkezet megújul.

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



8. A RUP tervezési munkafolyamatában (workflow) az architektúra tervezésekor milyen döntéseket hozunk ? (4 pont)

- Organize the system into packages (subsystems)
- Identifying concurrency
- Allocating packages to processors
- Storage and Persistence
- Handling global resources
- Choosing software control
- Handling boundary conditions

Eredmények értékelése:

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