

MI	Név, felvételi azonosító, Neptun-kód: MEGOLDÁS	pont(45) :
-----------	--	------------

Csak felvételi vizsga: <input type="checkbox"/>	csak záróvizsga: <input type="checkbox"/>	közös vizsga: <input type="checkbox"/>
---	---	--

Közös alapképzéses záróvizsga – mesterképzés felvételi vizsga Mérnök informatikus szak BME Villamosmérnöki és Informatikai Kar

2014. január 3.
MEGOLDÁSOK

A dolgozat minden lapjára, a kerettel jelölt részre írja fel nevét, valamint felvételi azonosítóját, záróvizsga esetén Neptun-kódját!

A fenti táblázat megfelelő kockájában jelölje X-szel, hogy csak felvételi vizsgát, csak záróvizsgát, vagy közös felvételi és záróvizsgát kíván tenni!

A feladatok megoldásához csak papír, írószer, zsebszámológép használata megengedett, egyéb segédeszköz és a kommunikáció tiltott. A megoldásra fordítható idő: 120 perc. A feladatok után azok pontszámát is feltüntettük.

A megoldásokat a feladatlpra írja rá, illetve ott jelölje. Teszt jellegű kérdések esetén elegendő a kiválasztott válasz betűjelének bekarikázása. Kiegészítendő kérdések esetén, kérjük, adjon világos, egyértelmű választ. Ha egy válaszon javítani kíván, teszt jellegű kérdések esetén írja le az új betűjelet, egyébként javítása legyen egyértelmű.

A feladatlpra írt információk közül csak az eredményeket vesszük figyelembe. Az áttekinthetetlen válaszokat nem értékeljük.

A vizsga végeztével mindenképpen be kell adnia dolgozatát. Kérjük, hogy a dolgozathoz más lapokat ne mellékeljen.

Felhívjuk figyelmét, hogy illegális segédeszköz felhasználása esetén a felügyelő kollegák a vizsgából kizárják, ennek következtében felvételi vizsgája, illetve záróvizsgája sikertelen lesz, amelynek letételét csak a következő felvételi, illetve záróvizsga-időszakban kísérelheti meg újból.

Szakirányválasztás

(Csak felvételi vizsga esetén kell kitölteni)

Kérem, az alábbi táblázatban jelölje meg, mely szakirányon kívánja tanulmányait folytatni. A táblázatban a szakirány neve mellett számmal jelölje a sorrendet: 1-es szám az első helyen kiválasztott szakirányhoz, 2-es a második helyen kiválasztotthoz tartozik stb. Nem kell az összes szakirány mellé számot írni, de legalább egy szakirányt jelöljön meg. Egy sorszám csak egyszer szerepeljen.

szakirány neve	gondozó tanszék	sorrend
Alkalmazott informatika szakirány	AAIT	
Autonóm irányító rendszerek és robotok szakirány	IIT	
Hálózatok és szolgáltatások szakirány	TMIT	
Hírközlő rendszerek biztonsága szakirány	HIT	
Intelligens rendszerek szakirány	MIT	
Médiainformatika szakirány	TMIT	
Rendszerfejlesztés szakirány	IIT	
Számításméletek szakirány	SZIT	
Szolgáltatásbiztos rendszertervezés szakirány	MIT	

AL	Név, felvételi azonosító, Neptun-kód: MEGOLDÁS	pont(15):
-----------	---	-----------

1. Legyen az $L(n)$ függvény értéke egy algoritmus maximális lépésszáma az n hosszú bemeneteken. Tudjuk, hogy $L(n) = L(n - 2) + \sqrt{n} + 5$. Következik-e ebből, hogy

a) $L(n) = O(n^2)$

igen — nem

b) $L(n) = O(\sqrt{n})$

igen — nem

pont(1):

2. Valamilyen bemenetet gyorsrendezéssel akartunk rendezni. Ha az algoritmus első menetének végrehajtása (az első particionálás) az alábbi számsort eredményezte, akkor mi lehetett ebben a menetben a kiválasztott elem? Az összes lehetőséget adja meg!

3, 5, 1, 7, 12, 8, 13, 15, 14

Megoldás: 7 vagy 13

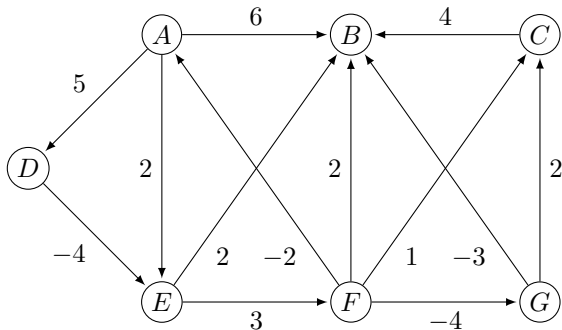
pont(1):

3. Hány különböző 4 élű út van a $K_{n,n}$ teljes páros gráfban, ha $n \geq 3$?

Megoldás: $n^2 \cdot (n - 1)^2 \cdot (n - 2)$

pont(2):

4. Az alábbi gráfon a Bellman–Ford-algoritmust használjuk az A pontból induló legrövidebb utak hosszának meghatározására. A kapott úthosszakat tartalmazó táblázat egyik sorát feltüntettük. Írja be a következő sorban keletkező értékeket!



	A	B	C	D	E	F	G
				...			
0	3	6	5	1	4	1	
0	-2	3	5	1	4	0	
				...			

pont(2):

5. Legyen $G = (V, E)$ egy irányított egyszerű gráf. A \mathcal{T} tulajdonság jelentse a következőt:

minden $k \geq 2$ szám, minden $u \in V$ és minden $x_1, x_2, \dots, x_k \in V$ esetén, ha $(u, x_1) \in E$ és $(x_i, x_{i+1}) \in E$ teljesül minden $i = 1, \dots, k - 1$ értékre, akkor $(x_k, u) \notin E$

Jellemezze szavakkal a \mathcal{T} tulajdonságú gráfokat! Milyen ismert algoritmussal dönthető el, hogy egy adott gráf rendelkezik-e a \mathcal{T} tulajdonsággal?

Megoldás: A gráfban nincs irányított kör. Algoritmus: DFS (az első visszaélig)

pont(2):

6. Egy városban a közlekedési csomópontok és a köztük levő útszakaszok egy irányítatlan egyszerű gráffal adottak. Egy stáb filmet akar forgatni a város utcáin. Mindenhol szívesen forgatnának, minden útszakaszra kialakítottak egy-egy árat, hogy mennyit fizetnének, ha a város azt az útszakaszt lezárátja a forgatás idejére. A város vezetése minél nagyobb bevételre akar szert tenni, de azt azért belátták, hogy minden utcát nem zárhatnak le. Ezért az a céljuk, hogy úgy határozzák meg a lezárandó útszakaszokat, hogy a fennmaradt részben is mindenholnan el lehessen jutni a városházára.

Melyik ismert algoritmus segíthet a megoldásban? Hogyan lehet annak eredménye alapján megadni a választ? (Az algoritmus részletes leírására nincs szükség.) *Megoldás:* Az úthálózat árákkal súlyozott gráfjában egy mi-

nimális súlyú feszítőfát keresünk (pl. Jarnik-Prim- vagy Kruskal-algoritmus). A fába nem kerülő éleknek megfelelő útszakaszokat zárják le.

pont(2):

7. Abból a feltevésből, hogy $P \neq NP$, az alábbiak közül melyik problémára következik, hogy nincs rá polinom idejű algoritmus?

\mathcal{A} : Adott egy G gráf. A csúcsai kiszínezhetők két színnel?

\mathcal{B} : Adott egy G gráf. A csúcsai kiszínezhetők 2014 színnel?

\mathcal{C} : Adott egy G gráf. A csúcsai kiszínezéséhez több mint 2014 szín kell?

Megoldás: \mathcal{B} és \mathcal{C}

pont(2):

8. Adott egy $n \geq 3$ csúcsú út, az i -edik csúcs súlyát jelölje s_i . Tegyük fel, hogy mindegyik s_i pozitív. A csúcsok egy X részhalmazának $s(X)$ súlya legyen az X -ben levő csúcsok súlyainak összege. Azt akarjuk meghatározni, hogy mekkora lehet az $s(X)$ legnagyobb értéke, ha X független csúcshalmaz.

Adjon egy lineáris idejű, dinamikus programozást használó eljárást erre a feladatra!

Megoldás:

$$T[1] = s_1$$

$$i = 2, 3, \dots, n$$

$$T[i] = \max(T[i - 1], T[i - 2] + s_i)$$

A keresett érték a $T[n]$.

pont(3):

H	Név, felvételi azonosító, Neptun-kód: MEGOLDÁS	pont(7,5):
----------	--	------------

Figyelem! Ha egy feladatnál több helyes válasz van, minden helyeset be kell jelölni!

1. Hogyan lehet használni a portszámokat az alkalmazások azonosítására?

- a) Mindig le kell kérdezni az alkalmazást, hogy milyen számú porton fogad üzeneteket.
- b) Sorsolni kell egy azonosítót, és azt szabadon lehet használni bármely alkalmazás elérésére.
- c) Az adatkapcsolati címek felhasználásával együtt.
- d) A fenti válaszok közül egyik sem helyes.

Megoldás: **d)**

pont(1):

2. Az alábbiak közül mely(ek) nem a TCP feladata(i)?

- a) Forgalomszabályozás
- b) Útvonalválasztás
- c) Sorrendhelyes átvitel
- d) Váltás a karakterkészletek között
- e) A fenti válaszok közül egyik sem helyes.

Megoldás: **b), d)**

pont(1):

3. Az alábbiak közül mely(ek) igaz(ak) a levelező rendszerekre?

- a) A felhasználók a User Agentekkel tudnak levelet küldeni és fogadni.
- b) Az e-mail címekben a @ utáni tag alapján DNS névfeloldás történik.
- c) A klienseknek a POP3 és IMAP4 protokollokat egyszerre használni kell a levelek letöltéséhez.
- d) A küldő fél MTA-ja addig tárolja a levelet, amíg a fogadó fél annak letöltését nem kezdeményezi.
- e) A fenti válaszok közül egyik sem helyes.

Megoldás: **a), b)**

pont(1):

4. Egészítse ki az alábbi állítást!

„A névfeloldás az Interneten a hálózati csomópont neve és annak IP-címe között teremt egyértelmű kapcsolatot. Ennek megvalósítását egy hierarchikus név- és címrendszer, a DNS végzi. E rend-

szerben a névfeloldási kérés elvileg először a hierarchia csúcsán álló-nak megy, az továbbítja a kérést az aktuális tartományba az azt kezelő name servernek.”

Megoldás: root

pont(1):

5. Egészítse ki az alábbi állítást!

A(z) egy olyan hálózatrész, amelyen belül egységes routing módszert alkalmaznak.

Megoldás: autonóm rendszer (AS)

pont(1):

6. Nevezze meg (magyarul vagy angolul) azt a jellemzően sokportos eszközt, amely a fizikai jeleket azok értelmezése nélkül továbbítja, és ezáltal több gép, illetve hálózat összekötését is lehetővé teszi!

Megoldás: hub

pont(1):

7. Egy IPv4-es hálózaton egy IP-fejrészrel együtt 2020 byte-os (opciókat nem tartalmazó) csomagot továbbítunk, mely során a csomag egy 1621 byte MTU-val rendelkező linken halad keresztül. Az átvitel során a csomagot a két link határán tördelik.

Mekkora a két töredék közül a második csomagtöredék mérete fejrészrel együtt? A teljes pontszámhoz a számítás menetét is írja le!

Megoldás: 420

pont(1,5):

O	Név, felvételi azonosító, Neptun-kód: MEGOLDÁS	pont(7,5):
----------	---	------------

Figyelem! Minden feladatnál csak egy helyes válasz van!

1. Az alábbi megállapítások közül melyik *hamis* a korai operációs rendszerekkel kapcsolatban?

- a) Az időosztásos rendszerek kötegelt (batch) feladatok futtatására is képesek.
- b) Az egyszerű monitorok (resident monitor) megjelenése előtt egy munka befejezése után az operátornak manuálisan kellett indítania a következő munkát.
- c) A korai operációs rendszerekben a programozási hibák keresése karakteres terminálon történt.
- d) A multiprogramozott operációs rendszerekben egy munka addig fut, ameddig várakozni nem kényszerül, ekkor az operációs rendszer egy másik munkát választ ki, és azt futtatja.

Megoldás: c)

pont(1):

2. Az alábbi állítások közül melyik *igaz* az operációs rendszerek programozói felületével kapcsolatban?

- a) Az alkalmazói programok szubrutinhívással érik el az operációs rendszer szolgáltatásait.
- b) Az alkalmazói programok egy operációs rendszer specifikus programkönyvtárat használnak az operációs rendszer szolgáltatásainak rendszerhívásokon keresztüli elérésére.
- c) A POSIX szabvány egységesen (szabványosan) megadja az operációs rendszer által nyújtandó rendszerhívások szintaxisát.
- d) A rendszerhívások processzorfüggetlenek, így biztosítják a programok hordozhatóságát.

Megoldás: b)

pont(1):

3. Az alábbi állítások közül melyik *hamis* az ütemezési algoritmusokkal kapcsolatban?

- a) A legrégebben várakozó (FIFO, FCFS) algoritmus esetén nagy lehet az átlagos várakozási idő (konvojhatás).
- b) A körforgó (RR) ütemező esetén ökölszabályként azt mondhatjuk, hogy a CPU löketek 80%-a hosszabb legyen az időszeletnél (time slice).
- c) A legrövidebb hátralevő idejű (SRTF) ütemező preemptív.
- d) A legrövidebb löketidejű (SJF) ütemezőben egy korábban beérkező nagy löketidejű feladat feltartja a végrehajtása alatt később beérkező kisebb löketidejű feladatokat (az átlagos várakozási idő nagy lehet).

Megoldás: b)

pont(1):

4. Az alábbi állítások közül melyik *hamis* a szála (thread)?

- a) A szálnak saját halma (heap) van.
- b) A szál magában szekvenciális kódot hajt végre.
- c) Egy operációs rendszerben csak egy adott folyamat kontextusában futó két szál között lehetséges a kommunikáció közös memória alkalmazásával.
- d) A folyamat létrejöttékor létrejön egy szál is a folyamat kontextusában.

Megoldás: a)

pont(1):

5. Az alábbi állítások közül melyik *hamis* folyamatokra (process)?

- a) A folyamat a CPU ütemezés alapegysége a modern operációs rendszerekben.
- b) Egy programból több független folyamat hozható létre és futtatható párhuzamosan.
- c) A folyamatot a kernelen belül a folyamatleíró (Process Control Block, PCB) írja le.
- d) A folyamatok közötti kommunikáció csak rendszerhívásokkal, az operációs rendszeren keresztül történhet.

Megoldás: **a)**

pont(1):

6. Mely állítás *igaz* a szemaforokkal kapcsolatban?

- a) A szemafor egy változó, amit tetszőlegesen kezelhetünk saját függvényeinkkel a programunkban.
- b) A P() művelettel szabadítjuk fel a szemaforral védett erőforrást.
- c) Feladatok szinkronizációjának (sorrendiségének) megvalósításához használt bináris szemafor szabad (1) értékre kell inicializálni.
- d) A szemafor belépési és kilépési műveletei oszthatatlanok (nem megszakíthatóak).

Megoldás: **d)**

pont(1):

7. Az alábbi virtuális tárkezeléssel kapcsolatos állítások közül melyik *igaz*?

- a) A virtuális tárkezelés során a fizikai memóriában nincs külső tördelődés.
- b) A virtuális tárkezelés során a teljes programot betöltjük a fizikai memóriába annak indulásakor.
- c) Ha a virtuális tárkezelés során egy folyamat érvénytelen címre hivatkozik, akkor azt hiba miatt leállítják.
- d) A virtuális memória sebességét a fizikai memória sebessége határozza meg.

Megoldás: **a)**

pont(1):

8. Az alábbi két állítás közül melyik *igaz*?

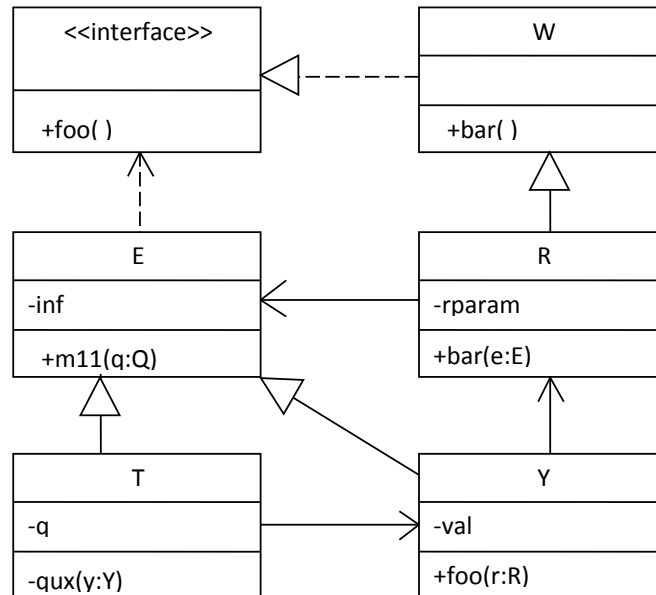
- a) A virtuális tárkezelés során az LRU (Legrégbben nem használt) lapcsere algoritmus esetén a frissen behozott lapokat a tárba kell fagyasztani.
- b) Perifériás műveletek (pl. DMA) célpontjaként megjelölt lapokat a tárba kell fagyasztani, mivel egyébként azok lecserélése esetén a perifériás műveletek hibát okozhatnának.

Megoldás: **b)**

pont(0,5):

S1	Név, felvételi azonosító, Neptun-kód: MEGOLDÁS	pont(5):
-----------	--	----------

1. Az alábbi UML2 diagram alapján – a kulcs felhasználásával – jellemezze az állításokat!



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

A T-nek m11(q:Q) metódusa kaphat paraméterként E-t, mert E a T őse.

Megoldás: **D**

pont(1):

2. Nevezzen meg egy statikus verifikációs technikát!

Megoldás: felülvizsgálat, átvizsgálás, review, audit

pont(1):

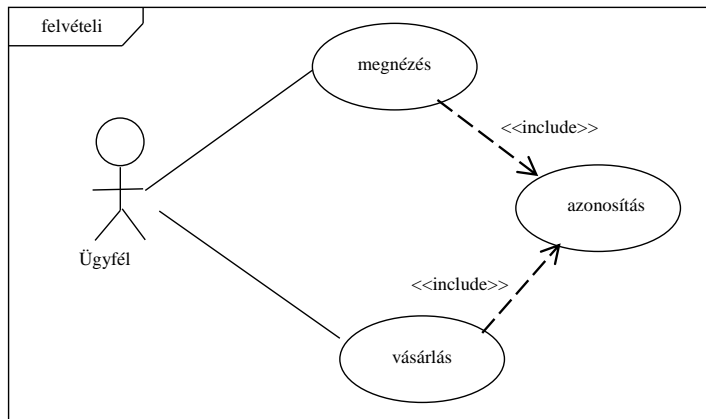
3. Az UML2 állapotdiagramjának (state chart) egy speciális esete egy másik UML2 diagram. Melyik ez a diagram?

Megoldás: Aktivitás diagram (Activity diagram)

pont(1):

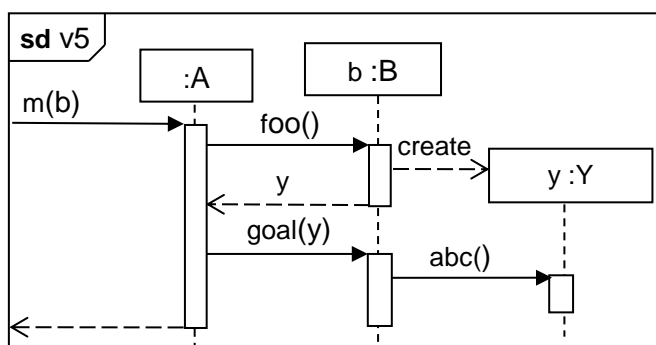
4. Egy ügyfél egy webáruházban vásárolhat, vagy megnézheti egy aktuális megrendelésének státuszát. Mindkét funkció igénybevételéhez azonosítania kell magát. Rajzoljon UML2 use-case (használati eset) diagramot!

Megoldás:



pont(1):

5. Tételezze fel, hogy az alábbi UML2 szekvenciadiagramon szereplő objektumok osztályai között nincs más egyéb – a diagramból nem kiolvasható – kapcsolat (pl. öröklés)! Mi a kapcsolat B és Y között?



- | | |
|-----------------------------------|---------------------------------------|
| a) asszociáció (association) | b) kollaboráció (collaboration) |
| c) interakció (interaction) | d) függőség (dependency) B függ Y-től |
| e) példányosítás (instantiation) | f) függőség (dependency) Y függ B-től |
| g) implementálás (implementation) | |

Megoldás: d)

pont(1):

S2	Név, felvételi azonosító, Neptun-kód: MEGOLDÁS	pont(5):
-----------	---	----------

1. Egy-két mondatban adja meg, milyen általános problémát old meg a Composite (Összetett) tervezési minta!

Megoldás:

- A rész-egész viszonyban álló objektumokat fastruktúrába rendezi.
- A kliensek számára lehetővé teszi, hogy a rész-egész viszonyban álló objektumokat egységesen kezelje.

pont(1):

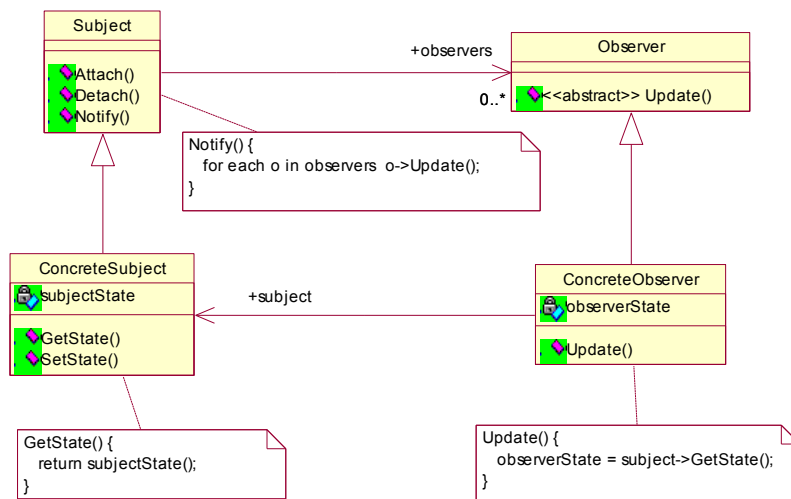
2. Milyen általános problémát old meg az Observer (Megfigyelő) tervezési minta?

Megoldás: Lehetővé teszi, hogy egy objektum a megváltozása esetén értesíteni tudjon tetszőleges más objektumokat anélkül, hogy bármit is tudna róluk.

pont(1):

3. Rajzolja fel az Observer minta osztálydiagramját, és jellemezze röviden az osztálydiagramon szereplő osztályokat!

Megoldás:



Subject : Tárolja a beregisztrált Observer-eket.

Observer: Interfészt definiál azon objektumok számára, amelyek értesülni szeretnének a Subject-ben bekövetkezett változásról.

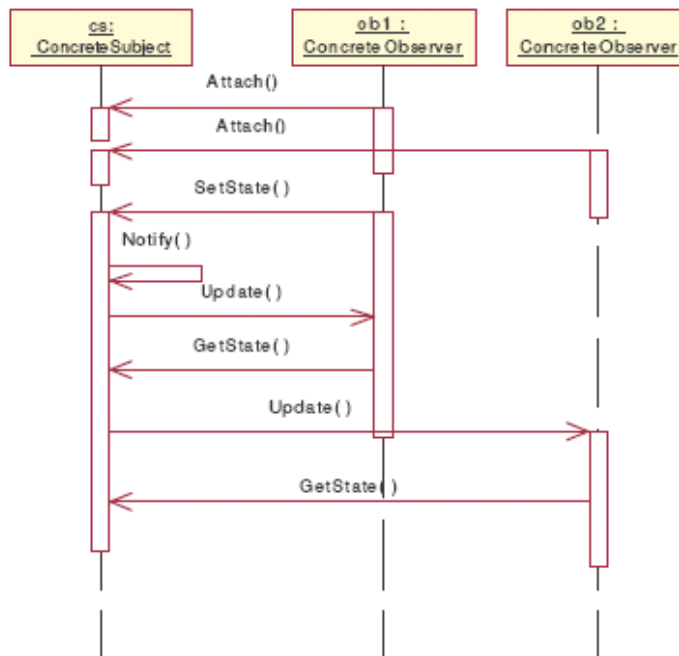
ConcreteSubject: Az observer-ek számára érdekes állapotot tárol, és értesíti a beregisztrált observer-eket, amikor az állapota megváltozik.

ConcreteObserver: Referenciát tárol a megfigyelt ConcreteSubject objektumra, implementálja az Observer interfészt (Update művelet).

pont(1):

4. Egy UML szekvenciadiagram segítségével mutassa be az Observer minta osztályainak együttműködését!

Megoldás:



A leszármazott ConcreteObserverek az Update függvény felülírásával értesülnek a Subject változásairól. Ilyenkor lekérik a ConcreteSubject állapotát, és reagálnak a változásra. Ha az egyik Observer változtatja meg a ConcreteSubject állapotát, akkor a Notify függvény meghívásával értesíthetik a többi Observert beleértve saját magukat is.

pont(1):

5. Tegyük fel, hogy egy adott művelet egy webalkalmazásban kliens (pl. JavaScript) és kiszolgáló (pl. ASPX) oldali kóddal is megvalósítható. Adjon meg egy előnyt a kliens oldali megvalósításra vonatkozóan, és egy tipikus előnyt a kiszolgáló oldali megvalósításra vonatkozóan!

Megoldás: A kliens oldali szkript (pl. JavaScript) előny pl.:

– Gyorsabb, mert nincs szükség interakcióra a kiszolgálóval (vagy ha szükség is van rá, az hatékonyabban, kisebb adatforgalom mellett megtehető).

A kiszolgáló oldali kód előnyök:

– A kiszolgáló oldali kód általában lefordítható. Így a hibák egy része már fordításkor kiderül, illetve az alkalmazás futása gyorsabb lesz.

– Kiszolgáló oldali kóddal általában könnyebb böngésző független megvalósítást készíteni.

pont(1):

AD	Név, felvételi azonosító, Neptun-kód: MEGOLDÁS	pont(5):
-----------	---	----------

1. Egy harangjáték-vezérlő adatbázisának relációs sémája: DALLAM(dalcím, dallamsor száma, ütem sorszáma, hangjegy sorszáma, hangmagasság, ritmusérték), ahol az első 4 attribútum összessége egy összetett kulcs. Relációalgebrai kifejezéssel adjuk meg a harangnak a lejátszandó hangokat (hangmagasság és ritmusérték párokat), de a harangjáték a várt dallam helyett összevissza hangsort szólaltat meg. Mi lehet a baj?

Megoldás: Az eredmény mindig egy halmaz és nem egy lista, függetlenül a lekérdezés módjától.

pont(1):

2. Hányadik normál formájú az $R(A,B,C,D)$ atomi attribútumokból álló relációs séma az alábbi függéshalmaz esetén?

$$F = \{B \rightarrow ACD, C \rightarrow D, A \rightarrow B, B \rightarrow C, D \rightarrow A\}$$

Megoldás: BCNF

pont(1):

3. Az alábbiak közül melyiket *nem* garantálják a BCNF sémák? Az összes helyes választ jelölje meg!

- a) összetett (nem atomi) attribútum létezését
- b) függőségőrzést (nemtriviális függéseket tartalmazó függéshalmaz esetén)
- c) ismétlődő attribútumértékek kizárását
- d) triviális funkcionális függés létezését
- e) másodlagos attribútum létezését
- f) értékfüggő kényszerek érvényesítését
- g) funkcionális függés alapú redundancia megszüntetését
- h) több kulcsot egy sémában

Megoldás: a), b), c), e), f), h)

pont(1):

4. Az $F = \{A \rightarrow B, A \rightarrow C, AC \rightarrow BC\}$ függéshalmazból a $G = \{AC \rightarrow B, AB \rightarrow C, A \rightarrow BC\}$ halmaz mely függései következnek?

Megoldás: Valamennyi

pont(1):

5. Mikor igaz egy funkcionális függés adott R séma attribútumain értelmezett F függéshalmaz mellett?

Megoldás: Ha minden olyan $r(R)$ reláción teljesül, amelyen F valamennyi függése fennáll.

pont(1):