

MI	Név, felvételi azonosító, Neptun-kód:	pont(90) :
-----------	---------------------------------------	------------

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

2011. 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	
Számításelemélet szakirány	SZIT	
Szolgáltatásbiztos rendszertervezés szakirány	MIT	

AL	Név, felvételi azonosító, Neptun-kód:	pont(30) :
-----------	---------------------------------------	------------

1. Legyen $f(n) = 2010\sqrt{n^5} + 8 \cdot 2^{2 \cdot \log_2 n} + 15$. Igaz-e, hogy

(i) $f = O(n^2)$?

Megoldás: nem

(ii) $f = O(n^3)$?

Megoldás: igen

pont(2):

2. Az alábbi hash-táblán a $h(x) = x \pmod{11}$ hash-függvénnyel kvadratikus maradékpróbát (álvéletlen próbálást) használva végezzük el a BESZŰR(19) műveletet. (A felső sor az indexeket mutatja.) A táblázat alá írja le, hogy a művelet után mi lesz az egyes mezőkben!

0	1	2	3	4	5	6	7	8	9	10
		24	13				7	18	30	
	19	24	13				7	18	30	

pont(2):

3. Egy 30 fős osztályban 3 ikerpár van, egy-egy ikerpár két tagja egyformán néz ki. A tanulók hány, látszólag is különböző módon tudnak sorba állni?

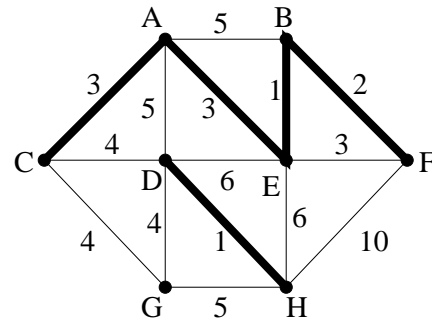
Megoldás: $\frac{30!}{2^3}$

pont(2):

4. Az alábbi gráfban minimális súlyú feszítőfát kezdünk el keresni.

(i) Ha a vastaggal jelölt élek már biztos részei a feszítőfának, akkor meg lehet-e állapítani, hogy Kruskal vagy Prim algoritmusát használjuk?

(ii) Hány különböző minimális feszítőfája van a gráfnak?



Megoldás: igen, Kruskal; 3 minimális feszítőfa van

pont(4):

5. Egy bináris keresőfában egész számokat tárolunk. A KERES(x) művelet során a fában érintett csomópontokban tárolt értékek sorban a következők voltak: -4, 100, 75, 21, -2, 7, 4. Mi lehet x értéke?

Megoldás: $-2 < x < 7$ egész szám

pont(4):

6. Az \mathcal{A} halmaz álljon az olyan $G = (V, E)$ irányítatlan egyszerű, összefüggő gráfokból, melyekre igaz a következő:

van olyan $k \geq 3$ egész és $v_0, v_1, \dots, v_{k-1} \in V$ különböző csúcsok, hogy $\{v_i, v_j\} \in E \Leftrightarrow j \equiv i \pm 1 \pmod{k}$

Jellemezze szavakkal az \mathcal{A} -beli gráfokat!

Megoldás: A gráfban van (feszített) kör.

pont(4):

7. Tegyük fel, hogy $P \neq NP$, és tekintsük a következő \mathcal{A} és \mathcal{B} problémát.

\mathcal{A} : Adott egy G irányítatlan gráf és egy $k > 0$ egész szám.
Kérdés, hogy van-e G -ben legalább k pontú kör.

\mathcal{B} : Adott egy G irányítatlan gráf és egy $k > 0$ egész szám.
Kérdés, hogy van-e a G gráfban két olyan csúcs, hogy minden őket összekötő út legalább k élből áll.

Van-e \mathcal{A} -ról \mathcal{B} -re, illetve \mathcal{B} -ről \mathcal{A} -ra Karp-redukció (polinomiális visszavezetés)? Válaszát röviden indokolja is!

Megoldás: $\mathcal{A} \in NP$ -teljes, $\mathcal{B} \in P$ tehát \mathcal{B} visszavezethető \mathcal{A} -ra de a feltétel miatt fordítva nem, azaz $\mathcal{B} \prec \mathcal{A}$ de $\mathcal{A} \not\prec \mathcal{B}$.

pont(6):

8. Egy súpályarendszeren minden pálya két megadott csomópont között vezet. Tudjuk, hogy minden pálya az egyik irányban lejt. Legyenek adottak a csomópontok, és minden pályára, hogy melyik kettő csomópontot köti össze, továbbá, hogy melyik irányba lejt és mennyi a pálya hossza. Egymáshoz csatlakozó pályák olyan sorozatát keressük, hogy ezeken végighaladva mindig lefele sielhessünk. Vázzon egy algoritmust, mellyel meg lehet határozni a feltételnek megfelelő leghosszabb útvonal hosszát, és hogy ehhez melyik csomópontból kell indulni! Az algoritmus lépésszáma c darab csomópont és p darab pálya esetén legyen $O(c + p)$.

Megoldás: A pályarendszer egy c pontú p élű irányított gráffal írható le. Ebben a gráfban nincs irányított kör (dag). A feladat a gráfban leghosszabb út keresése. A PERT módszer $O(c + p)$ lépésben meghatározza minden pontra az onnan induló leghosszabb utat. A kapott hosszak maximumát keresve megkapjuk, melyik csomópontból kell indulni. Ez csak további $O(c)$ lépés, tehát összesen $O(c + p)$.

(PERT helyett topologikus rendezés + dinamikus programozás is jó.)

pont(6):

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

1. Mely(ek) az adatkapcsolati rétegben működő, hálózatrészeket összekapcsoló átjáró(k)?

- a) híd (bridge)
- b) kapcsoló (switch)
- c) útválasztó (router)
- d) többkapus jelismétlő (hub)
- e) jelismétlő (repeater)
- e) átjáró (gateway)

Megoldás: **a), b)**

pont(2):

2. Hányszorosára nő a bitsebesség, ha kétszintű ASK-ról BPSK-ra váltunk azonos szimbólumsebesség mellett? (A válasz egy szám, melyben szükség esetén tizedespont használható.)

Megoldás: 1

pont(2):

3. Összehasonlítva az egyszerű Alohával, a réselt Alohára az alábbiak közül mely állítás(ok) igaz(ak)?

- a) A kihasználtság javul az egyszerű Alohához képest.
- b) A késleltetés nagymértékben javul az egyszerű Alohához képest.
- c) A fairness elvek sokkal jobban érvényesülnek, mint az egyszerű Alohánál.
- d) Mai napig használják egyes rendszerekben.

Megoldás: **a), d)**

pont(2):

4. Egészítse ki az alábbi állítást!

A 802.11 hálózatoknak azt a működési módját, amikor csak a kliensek vesznek részt a kommunikációban, peer-to-peer módon, módnak nevezzük.

Megoldás: ad hoc

pont(2):

5. Miért nem lehet minden esetben egyetlen adatkapcsolati rétegbeli keretben továbbítani egy IP csomagot?

- a) Mert biztonságosabb, ha részekben továbbítjuk.
- b) Mert gyorsabban ér célba, ha kisebb részekre tördelve továbbítjuk.
- c) Mert ez javítja a csatorna kihasználtságát.
- d) Mert az adatkapcsolati protokollok nem mindig engednek meg akkora adatmennyiséget a keretben, mint amekkora egy IP csomag lehet.
- e) Egyik felsorolt válasz sem jó.

Megoldás: **d)**

pont(2):

6. Egészítse ki az alábbi állítást!

A névfeloldás az interneten a hálózati csomópont neve és annak $\overset{1}{\text{.....}}$ -címe között teremt egyértelmű kapcsolatot. Ennek megvalósítását egy hierarchikus név- és címrendszer, a $\overset{2}{\text{.....}}$ végzi. A terhelés csökkenthető másodlagos kiszolgálók (szerver-replikáció) segítségével, vagy $\overset{3}{\text{.....}}$ alkalmazásával. Utóbbi esetben, ha a végpont vagy ügynöke felold egy nevet, akkor ennek az eredménye tárolásra kerül az ismételt kérések kiszolgálásának felgyorsítása végett.

A 3 hiányzó elnevezés:

- 1) ... IP
- 2) ... DNS
- 3) ... cache

pont(2):

7. Az *A* és *B* végpont közötti kommunikáció során *A* végpont által küldött utolsó 4 szegmens TCP PDU-jában a sorszám (sequence number) 178, 190, 222, 253. *B* válaszként küldött 3 szegmens TCP PDU-jában az ACK-szám 210, 210, 210. Erre válaszként az *A* terminal milyen szekvenciájú és hány bájtos szegmenst fog küldeni fast retransmit esetén, amennyiben nem járt még le a time-out idő?

Megoldás: 222, 31 byte

pont(3):

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

1. Az alábbi állítások közül melyik *nem igaz* az operációs rendszerekkel kapcsolatban?

- a) Az operációs rendszer egyik feladata a hardver-erőforrások felügyelete.
- b) Az alkalmazói programok az operációs rendszer által nyújtott környezetben futnak.
- c) Az operációs rendszer a felhasználói programok számára egy virtuális gépet hoz létre, elfedve a hardver specialitásait.
- d) A kemény valós idejű (hard real-time) operációs rendszerek az emberi reakcióidőhöz képest gyorsan válaszolnak a hozzájuk intézett kérésekre.

Megoldás: **d)**

pont(2):

2. Az alábbi állítások közül melyik *nem igaz* az operációs rendszerek tipikus belső felépítésével kapcsolatban?

- a) Az operációs rendszerek forráskódjában a hardverspecifikus kódrészletek külön szoftvermodulokba kerülnek.
- b) Az eszközmeghajtók (device driver) feladata a hardver és az operációs rendszer közötti kapcsolat megteremtése egy, az operációs rendszer számára egységes (szabványos) interfészen keresztül.
- c) A felhasználói programok ki- és bemeneti gépi utasításokkal elérhetik a hardver elemeket.
- d) Az operációs rendszerek belső felépítése réteges.

Megoldás: **c)**

pont(2):

3. Az alábbi állítások közül melyik *igaz* a folyamatok (process) vagy szálak (thread) témakörében?

- a) A szál egy végrehajtás alatt álló program.
- b) A folyamatoknak saját verme (stack) van, míg a szálaknak ilyen nincs, azok osztoznak az őket magában foglaló folyamat vermén.
- c) A szálak osztoznak az őket magukba foglaló folyamat halmán (heap).
- d) A szálak fő előnye, hogy azok között csak rendszerhívásokkal lehet kommunikálni, így azok jobban védve vannak egymástól.

Megoldás: **c)**

pont(2):

4. Mely állítás *nem igaz* PRAM modell alapján működő memóriahozzáférés esetén?

- a) A PRAM modell párhuzamosan futó folyamatok esetén határozza meg a memória működését.
- b) A PRAM modell szerint olvasás-írás ütközés esetén a rekesz tartalma felülíródik a beírni szándékozott értékkel, az olvasás pedig vagy a rekesz régi, vagy az új értéke lesz, más érték olvasása nem lehetséges.
- c) A PRAM modell írás-írás ütközés esetén hibát jelez.
- d) A PRAM modell esetén ütközésnél lényegében valamilyen előre nem meghatározható sorrendben, de mindig helyesen történnek meg a párhuzamosan végrehajtott memóriaműveletek.

Megoldás: **c)**

pont(2):

5. Mely állítás *hamis* az alábbi virtuális tárkezeléssel kapcsolatos megállapítások közül?

- a) Virtuális tárkezelés esetén a futó programok méretét nem korlátozza a rendelkezésre álló fizikai memória mérete.
- b) A gyakori laphibák által okozott teljesítménycsökkenést vergődésnek (trashing) nevezzük.
- c) Az előrettekintő lapozás (anticipatory paging) mindig jobb teljesítmény nyújt, mint a igény szerinti lapozás (demand paging).
- d) A legrégebbi lap (FIFO) algoritmus esetén megjelenhet a Bélády-anomáliának nevezett jelenség, amikor is a folyamathoz tartozó fizikai memóriakeretek számának növelésével a laphibák gyakorisága nőhet.

Megoldás: **c)**

pont(2):

6. Melyik állítás *igaz* az alábbiak közül az állományok és azok lemezen történő tárolása tekintetében?

- a) A lemezen csak az egyes állományokhoz tartozó blokkok nyilvántartásával kell törődnünk.
- b) A láncolt listás tárolás lassú, hiszen a teljes listát az elejétől végig kell járnunk egy, a listában elhelyezkedő elem megtalálásához.
- c) A FAT fájlrendszer indexelt tárolást használ.
- d) A láncolt listás tárolás belső tördelődéstől mentes.

Megoldás: **b)**

pont(2):

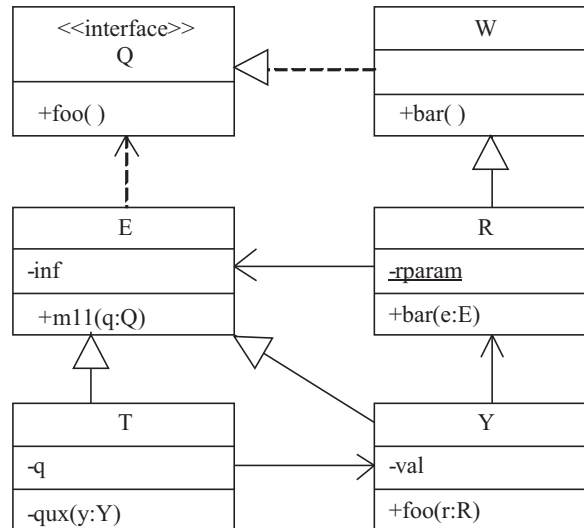
7. Vázolja ábrával egy folyamat állapotátmeneti diagramját a legegyszerűbb esetben! Adja meg preemptív és nem preemptív ütemezők esetén meglévő különbségeket az ábrán!

Megoldás: könyv 141. oldal, 3.5 ábra.

pont(3):

S1	Név, felvételi azonosító, Neptun-kód:	pont(10) :
-----------	---------------------------------------	------------

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 (- -)

(i) Y m11(q:Q) metódusa kaphat paraméterül R objektumot, mert Y függ R-től.

(ii) Y foo(r:R) metódusa nem módosíthatja a paraméter rparam attribútumát, mert az attribútum statikus.

Megoldás: **B, B**

pont(2):

2. Mi az architektúrális tervezés célja?

Megoldás: A specifikált rendszer fő komponenseinek azonosítása és a közöttük fennálló együttműködés (kommunikáció) definiálása.

pont(2):

3. Jelölje meg, hogy az alábbi kérdések mely fogalmakhoz kapcsolódnak!

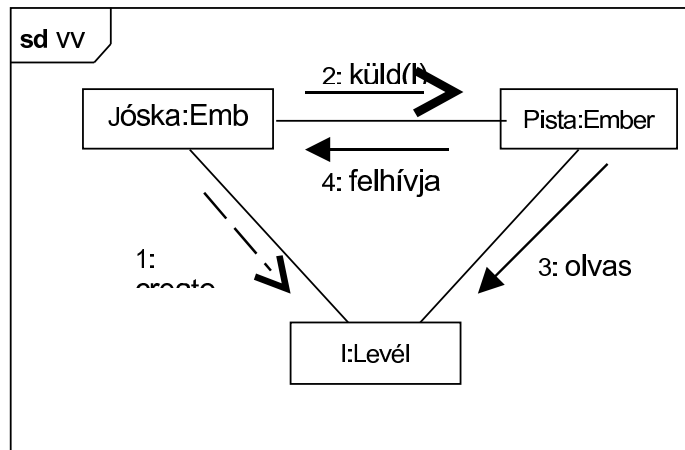
(i) Ez a jó szoftver? **Verifikáció – Validáció**

(ii) Jó ez a szoftver? **Verifikáció – Validáció**

pont(2):

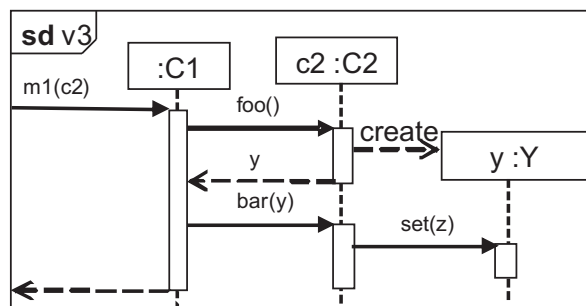
4. Jóska az öccse, Pista névnapjára levelet ír, amelyet elküld neki. Pista a levelet azonnal elolvassa, és rögtön felhívja bátyját telefonon. Rajzoljon UML2 kommunikációs diagramot!

Megoldás:



pont(2):

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 C1 és C2 között?



- a) példányosítás (instantiation)
- b) asszociáció (association)
- c) kollaboráció (collaboration)
- d) függőség (dependency) C1 függ C2-től
- e) függőség (dependency) C2 függ C1-től
- f) interakció (interaction)
- g) implementálás (implementation)

Megoldás: d)

pont(2):

S2	Név, felvételi azonosító, Neptun-kód:	pont(10) :
-----------	---------------------------------------	------------

1. Egy-két mondatban adja meg, milyen általános problémát old meg a Proxy (Helyettes) tervezési minta?

Megoldás: Objektum helyett egy helyettesítő objektumot használ, ami szabályozza az objektumhoz való hozzáférést.

pont(2):

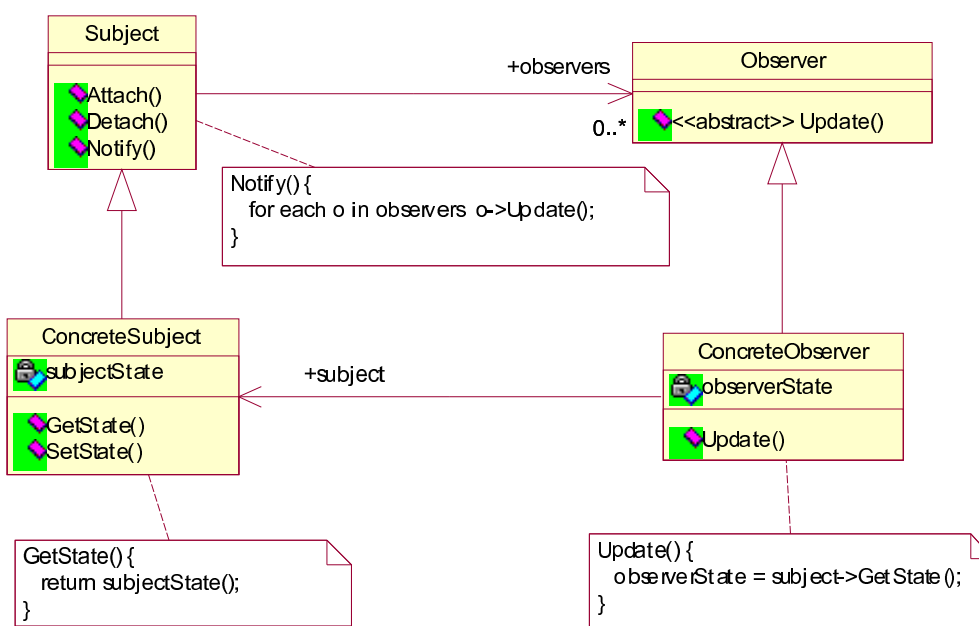
2. Milyen általános problémát old meg az Observer (Megfigyelő) tervezési minta?

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

pont(2):

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.

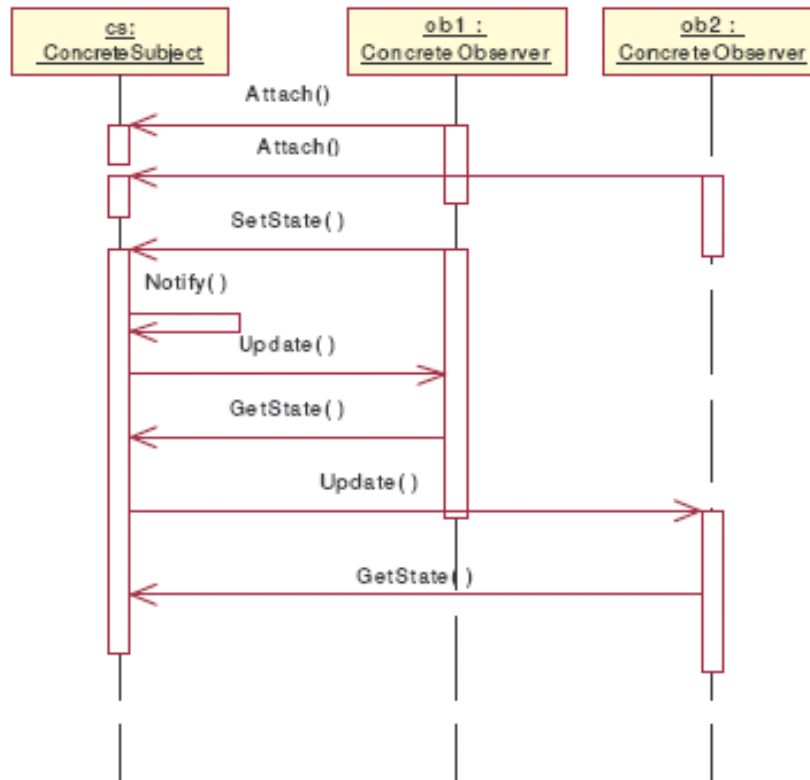
ConcreteSubject: Az observer-ek számára érdekes állapotot tárol, és értesíti a beregisztrált observer-eket, amikor az állapota megváltozik.

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

pont(2):

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

Megoldás:



A lezármazott 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(2):

5. Hasonlítsa össze a kliens és a kiszolgáló oldali szkript szerepét a webalkalmazásokra vonatkozóan!

Megoldás: A kliens oldali szkript (pl. JavaScript) a böngészőben fut, a felhasználó számára megjelenített HTML oldal tartalmát és a böngésző megjelenését módosítja (pl. ablak mozgatás, új ablak megnyitás, stb.). A kiszolgáló oldali szkriptet a webkiszolgáló futtatja, szerepe egyrészt a beérkező kérés feldolgozása (pl. a felhasználó által megadott adatok kinyerése), másrészt a kliens számára visszaküldött HTML oldal előállítás. Ehhez a kiszolgáló oldali szkript felhasználhatja a szerver oldali erőforrásokat is (pl. adatbázis).

pont(2):

AD	Név, felvételi azonosító, Neptun-kód:	pont(10):
-----------	---------------------------------------	-----------

1. Egy állomány rekordjait sűrű index, majd erre épített egyszintes ritka index segítségével szeretnénk elérni. Adjon alsó becslést a tároláshoz szükséges blokkok számára, ha az állomány $3 \cdot 10^6$ rekordból áll, egy rekord hossza 300 byte, egy blokk hasznos mérete 3000 byte, a kulcs 45 byte, egy mutató 5 byte!

Megoldás: 1 blokkban elfér 10 adatrekord, ill. 60 indexrekord. Az adatblokkok száma tehát legalább 300 000. A sűrű index rekordok száma is $3 \cdot 10^6$, ehhez legalább $3 \cdot 10^6 / 60 = 50\,000$ (sűrű) indexblokk kell. A ritka index rekordok száma legalább 50 000, ehhez legalább $50\,000 / 60 = 8334$ blokk kell. Összesen: $300\,000 + 50\,000 + 8334 = 358\,334$ adatblokk kell legalább.

pont(2):

-
2. Igaz-e, hogy nemtriviális függésekből a tranzitivitási axióma alkalmazásával kapott függés mindig tranzitív?

Megoldás: nem igaz

pont(2):

-
3. Az $R(ABCDEFG)$ séma mely attribútumaiban valósulhat meg redundáns adattárolás az

$$F = \{A \rightarrow BC, DA \rightarrow G, D \rightarrow EF, E \rightarrow F, F \rightarrow E\}$$

függéshalmaz mellett funkcionális függőség következtében?

Megoldás: B, C, E, F

pont(2):

-
4. Melyik legmagasabb normál formában van az atomi attribútumokat tartalmazó $R(A, B, C, D)$ relációs séma, ha

$$F = \{C \rightarrow B, B \rightarrow D, AB \rightarrow AC, CD \rightarrow B\} ?$$

Megoldás: 1NF

pont(2):

-
5. Ekvivalens-e a következő két függéshalmaz?

$$\{AB \rightarrow C, AC \rightarrow B, A \rightarrow BC\} \quad \text{és} \quad \{A \rightarrow B, A \rightarrow C, AC \rightarrow BC\}$$

Megoldás: igen

pont(2):