

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

Csak felvételi vizsga: ☐	csak záróvizsga: ☐	közös vizsga: ☐
---	---	--

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

2017. május 23.
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 feladatlagra 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 feladatlagra 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.

Specializációválasztás (Csak felvételi vizsga esetén kell kitölteni)

Kérem, a túloldalon található táblázatokban jelölje meg, mely fő-, illetve mellékspecializáción kívánja tanulmányait folytatni. FIGYELEM! A fő- és mellékspecializációkat külön-külön kell sorrendbe állítani!

Főspecializáció választása
(Csak felvételi vizsga esetén kell kitölteni)

A táblázatban a főspecializáció neve mellett számmal jelölje a sorrendet: 1-es szám az első helyen kiválasztott specializációhoz, 2-es a második helyen kiválasztotthoz tartozik stb. Nem kell az összes főspecializáció mellé számot írni, de legalább egy főspecializációt jelöljön meg.

Főspecializáció	sorrend
Alkalmazott informatika (AUT)	
Internetarchitektúra és szolgáltatások (TMIT)	
Kritikus rendszerek (MIT)	
Mobil hálózatok és szolgáltatások integrációja (HIT)	
Vizuális informatika (IIT)	

Mellékspecializáció választása
(Csak felvételi vizsga esetén kell kitölteni)

A táblázatban a mellékspecializáció neve mellett számmal jelölje a sorrendet: 1-es szám az első helyen kiválasztott specializációhoz, 2-es a második helyen kiválasztotthoz tartozik stb. Nem kell az összes mellékspecializáció mellé számot írni, de legalább egy mellékspecializációt jelöljön meg.

Mellékspecializáció	sorrend
Adat- és médiainformatika (TMIT)	
IT biztonság (HIT)	
IT rendszerek fizikai védelme (HVT)	
Intelligens rendszerek (MIT)	
Mobilszoftver-fejlesztés (AUT)	
Számításelmélet (SZIT)	
Számítási felhők és párhuzamos rendszerek (IIT)	

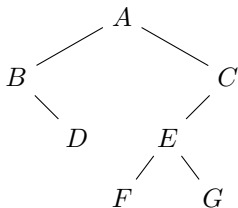
AL	Név, felvételi azonosító, Neptun-kód: MEGOLDÁS	pont(15):
-----------	---	-----------

1. Legyen $f(n) = f(n-3) + 8$, ha $n > 1$ és $f(1) = 2015$, $f(2) = 2016$, $f(3) = 2017$. Melyik az a legkisebb pozitív egész d szám, melyre $f(n) = O(n^d)$ teljesül?

Megoldás: 1

pont(1):

2. Adja meg, hogy az alábbi bináris fa esetén mi lesz a csúcsoknak az inorder bejárás szerinti sorrendje!



Megoldás: B, D, A, F, E, G, C

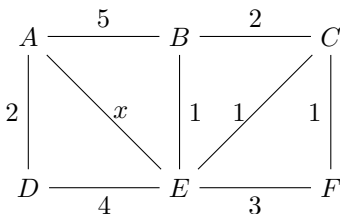
pont(1):

3. 100 ember közül kell kiválasztanunk, hogy két különböző feladat közül ki melyiket csinálja és esetleg ki maradjon feladat nélkül (mindenki legfeljebb az egyik feladathoz lesz rendelve). Az egyik feladatra pontosan 10 ember kell, a másikra a többiek közül tetszőleges számú beosztható, de legalább 1 fő kell (és maradhatnak feladat nélküliek is). Hányféle beosztás lehetséges? (Nem konkrét számot, hanem egy formulát várunk.)

Megoldás: $\binom{100}{10} \cdot (2^{90} - 1)$

pont(2):

4. Az alábbi gráfon a Kruskal-algoritmus az $\{A, E\}$ élet nem választotta be a minimális feszítőfába. Mi lehetett a negyediként beválasztott él, és milyen lehetőségek vannak az x értékére, ha tudjuk, hogy x egy egész szám?



Megoldás: Az $\{A, D\}$ él a negyedik, $x \geq 4$

pont(2):

5. Adottak az $A[i,j]$ nemnegatív egész számok, $1 \leq i, j \leq n$.

A $\mathcal{T}$ tulajdonság a következő: $\sum_{j=1}^n \prod_{i=1}^n A[i,j] = 0$ és $\sum_{i=1}^n \prod_{j=1}^n A[i,j] = 0$.

Fogalmazza meg, milyen tulajdonságot jelent ez az $A[i,j]$ elemeket tartalmazó, $n \times n$ méretű A tömbre!

Megoldás: A tömb minden sorában és oszlopában van 0.

pont(2):

6. Tegyük fel, hogy $P \neq NP$. Tekintsük a következő eldöntési problémákat:

$\mathcal{A}$: Adott egy G irányítatlan gráf.

Van-e G -ben Hamilton-kör?

$\mathcal{B}$: Adott egy G irányítatlan gráf.

Kiszínezhetők-e G csúcsai 4 színnel úgy, hogy pontosan 3 csúcs legyen piros színű?

$\mathcal{C}$: Adott egy G irányítatlan gráf.

Kiszínezhetők-e G csúcsai 3 színnel úgy, hogy pontosan 4 csúcs legyen piros színű?

Van-e Karp-redukció (polinomiális visszavezetés)

$\mathcal{A}$ -ról a $\mathcal{C}$ -re? igen – nem

$\mathcal{B}$ -ről a $\mathcal{A}$ -ra? igen – nem

$\mathcal{C}$ -ről a $\mathcal{A}$ -ra? igen – nem

pont(2):

7. Egy kommunikációs hálózat egy irányítatlan gráffal van megadva. Ebben minden élhez tartozik egy $0 \leq s \leq 9$ egész szám, ami az adott kapcsolat biztonsági szintjét jelzi (9, ha szuper biztonságos, 0 ha egyáltalán nem biztonságos). Egy útvonal biztonsági szintje a benne szereplő legkisebb biztonsági szinttel egyenlő.

Olyan algoritmust szeretnénk, aminek, ha megadunk két csúcsot (A és B) és egy T biztonsági szintet, akkor megadja, hogy van-e legalább T szintű útvonal A és B között, és ha van ilyen, akkor minimálisan hány csomóponton kell az üzenetnek áthaladnia.

Melyik ismert algoritmussal, azt milyen bemeneten használva lehet polinomidőben megadni a választ?

Megoldás: A gráfból csak a legalább T szintű éleket tartjuk meg. Ezen egy BFS megadja az elérhetőséget és a legkevesebb élből álló jó utat is.

pont(2):

8. Adottak az $s_1, s_2, \dots, s_n$ pozitív egészek, és egy b pozitív egész. Definiálunk egy T tömböt, melynek n sora és $b+1$ oszlopa van. Legyen $T[i, 0] = 1$ minden $1 \leq i \leq n$ esetén. A tömb további részében, $1 \leq i \leq n$ és $1 \leq c \leq b$ esetén $T[i, c]$ legyen az a szám, amely megadja, hogy hányféleképpen lehet az $s_1, s_2, \dots, s_i$ közül néhány összegeként a c számot előállítani (az összegben mindegyik s_i legfeljebb egyszer használható).

Milyen sorrendben és milyen formula alapján lehet kitölteni a tömböt?

Megoldás: Pl. soronként, $T[1, c] = 1$ ha $a_1 = c$, egyébként 0.

$i = 2, \dots, n$: ha $c < a_i$, akkor $T[i, c] = T[i-1, c]$ egyébként pedig $T[i, c] = T[i-1, c] + T[i-1, c-a_i]$.

pont(2):

Hogyan lehet a kitöltött táblázat alapján meghatározni, hogy mely $c \leq b$ számok állíthatók elő néhány s összegeként?

Megoldás: Azok a c számok, melyekre $T[n, c] > 0$.

pont(1):

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

1. Egészítse ki az alábbi szöveget!

Az AS egy olyan hálózatrész, amelyen belül egységes routing módszert alkalmaznak. Ezek között az EGP csoportba tartozó routing protokollokat használhatjuk, melyeknek leggyak-

rabban használt változata a(z)

Megoldás: BGP

pont(1):

2. A 10.1.128.0/22 hálózatban adja meg a router IP címét decimális formában, amennyiben a router a legnagyobb kiosztható IP címet kapta!

Megoldás: 10.1.131.254

pont(1):

3. Egészítse ki az alábbi állítást!

A HTTP számára a TCP-nek az a szerepe, ami az RTP számára a(z)

.....-nak/-nek.

Megoldás: UDP

pont(1):

4. DNS névfeloldásnál hogy nevezzük azt a választ, amelyik a rekordért felelős szerverek valamelyikétől származik?

Megoldás: autoritativ válasz

pont(1):

5. Válassza ki a következő többszörös hozzáférési módszerek közül a központilag vezéreltet!

- a) CSMA/CA;
- b) csoportos lekérdezés (probing);
- c) Réselt Aloha;
- d) bit-térkép módszer.

Megoldás: b)

pont(1):

6. Hogy nevezzük azt a címzési módot IP hálózatokban, ahol a csomag egy csoport összes tagjának szól?

Megoldás: multicast

pont(1):

7. Számolja ki a pillanatnyi adatsebességet Mbit/s-ban TCP átvitel során egy csomagvesztés után, amennyiben AIMD torlódásvezérlési eljárást használunk és a körülfordulási idő 300 ms, a csomagvesztési időpont előtti torlódási ablak (congestion window) pedig 900 kByte volt!

Megoldás: 12 Mbit/s

pont(1,5):

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

Figyelem! Minden feladatnál csak egy helyes válasz van!

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

- a) Az operációs rendszert egy felhasználói program futása közben csak egy megszakítás (rendszerhívás, HW megszakítás vagy kivétel) aktiválhatja.
- b) A rendszerhívás mindig visszatér az azt hívó felhasználói programba.
- c) Az aszinkron rendszerhívások csupán elindítják a kért rendszerfunkció végrehajtását, a tényleges eredmények kezelésére más mechanizmusok állnak rendelkezésre.
- d) A be- és kiviteli műveleteket rendszerhívásokon keresztül érhetik el a felhasználói programok.

Megoldás: **b)** (mivel számos rendszerhívás soha nem tér vissza, pl. `TerminateProcess()`).

Az **a)**, **c)** és **d)** megoldások definíciószerűen igazak.)

pont(1):

2. Az alábbi állítások közül melyik *hamis* az ütemezési algoritmusok jellemzésére használt mértékekre?

- a) Az áteresztőképesség mértékegysége az 1/s vagy job/s.
- b) Az átlagos körülfordulási idő mindig nagyobb, mint az átlagos várakozási idő.
- c) A központi egység kihasználtsága nem lehet 100%-nál több egyprocesszoros rendszerben.
- d) A CPU-kihasználtság definíciószerűen a CPU bármilyen munkával töltött idejének aránya az összes időhöz képest.

Megoldás: **d)** (mert csak a hasznos munkát vesszük figyelembe, az adminisztratív feladatokat nem.)

pont(1):

3. Az alábbi állítások közül melyik *igaz* a következő egyszerű ütemezési algoritmusokkal (FIFO, RR, SJF, SRTF) kapcsolatban?

- a) A SJF algoritmus nem preemptív.
- b) Az RR algoritmus nem preemptív.
- c) Az SRTF algoritmus alkalmazása esetén jelentkezhet a konvojhatás.
- d) A FIFO algoritmus esetén a futó feladat megszakítás nélkül, befejezésig fut.

Megoldás: **a)** (mert az SJF nem preemptív.

b) Az RR preemptív, **c)** az SRTF esetén nem jelentkezik a konvojhatás, mert az preemptív, **d)** a FIFO esetén az I/O várakozást okozhat)

pont(1):

4. Az alábbi állítások közül melyik *hamis* a folyamatokkal (process) és a szálakkal (thread) kapcsolatban egy folyamatokat és azokon belül szálakat támogató operációs rendszerben?

- a) A szálak osztoznak a folyamat memóriáján
- b) A folyamathoz a programozónak kell az alapértelmezett szálát hozzárendelnie.
- c) A szálnak saját verme (stack) van.
- d) Az ütemező többnyire szálakat ütemez.

Megoldás: **b)** (az alapértelmezett szálát a rendszer létrehozza, pl. abban futtatja a `main()` vagy azzal ekvivalens függvényt

a) A folyamat egy memória konténer lényegében, **c)** Egyébként nem tudna saját függvényt hívni a szál, **d)** igen, bár egyes rendszerek, pl. Linux, generikusan kezel ütemezhető objektumokat, amik lehetnek szálak is (és szinte kivétel nélkül azok is).)

pont(1):

5. Az alábbi állítások közül melyik *hamis* a mutex-ra vonatkozóan?

- a) A mutex elsősorban többpéldányos közös erőforrásokra vonatkozó kölcsönös kizárás megvalósítására szolgál.
- b) A mutex működése lényegében azonos az 1 értékkel inicializált „bináris” szemafor működésével.
- c) A mutex az operációs rendszerben kerül megvalósítására, azt rendszerhívásokkal lehet elérni.
- d) A mutex a modern operációs rendszerekben többnyire ESEMÉNYRE VÁR állapotba helyezi az azt elérni kívánó feladatot, ha a mutex foglalt.

Megoldás: a) (mert a mutex egypéldányos erőforrások esetén alkalmazható.

b) igaz, mert a két megoldás működése ebben az esetben teljesen azonos, c) egyébként nem tudná elvégezni a feladatát,

d) igen, az erőforrásra ebből következően passzívan várakozunk (sleeplock).)

pont(1):

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

- a) A virtuális tárkezelés esetén lehet külső tördelődés a fizikai memóriában.
- b) A virtuális tárkezelés alapja a fizikai memória szegmensekre bontása.
- c) A vergődés (trashing) elkerülésére a folyamat rendelkezésére álló virtuális memóriát kell növelni.
- d) A folyamatok laptáblái a fizikai memóriában találhatók.

Megoldás: d) (mert az MMU-nak is el kell érnie és az OS-nek is, valamint minden folyamatnak saját laptáblája van, vagyis nagyon máshol nem lehet eltárolni. A TLB ennek egy gyorsító tára, ez már az MMU-ban lévő speciális memória, de a kérdés nem erre vonatkozik.

Az a) rossz, hiszen a lapozás lényege, hogy nincs küldő tördelődés. A b) esetén a lapozás a helyes válasz, a szegmentálás ettől teljesen független fogalom. c) A folyamathoz tartozó fizikai memóriát kell növelni, a virtuális memória adott, rendszerfüggő, a folyamattól nem függ (max. attól, hogy az egy 32 bites vagy 64 bites alrendszerben fut).)

pont(1):

7. Az alábbi állítások közül melyik *igaz* a láncolt listás és az indexelt tároláson alapuló fájlrendszer esetén?

- a) A láncolt listás megoldás esetén a listát leíró információt speciális, ún. lista blokkokban tároljuk.
- b) Az indexelt tárolás esetén az indexet leíró információt nem kell védeni adatvesztés ellen (pl. replikálni), a fájl tartalma indexblokkok elvesztése esetén is elérhető.
- c) Az indexelt tárolás kedvezőbb mind szekvenciális, (fájl elejétől végéig olvasása) mind közvetlen (információ olvasása a fájl belsejéből) elérésre.
- d) A láncolt listás tárolás kedvező mind szekvenciális (fájl elejétől végéig olvasása) mind közvetlen (információ olvasása a fájl belsejéből) elérésre.

Megoldás: c) (mert mindkét esetben csak az index blokkok olvasásával megtalálható az éppen szükséges információ.

Az a) hibás, hiszen itt a blokkokban helyezzük el a láncolt lista leírásához szükséges információt és a fájl tartalmát is. A

b) hibás, hiszen az index blokkok elvesztése a fájl elvesztését jelenti, vagyis az indexet védeni kell adatvesztés ellen. A láncolt lista kedvezőtlen közvetlen hozzáférés esetén.)

pont(1):

8. Az alábbi két állítás közül melyik *hamis* az erőforráshasználat miatt kialakuló holtponttal kapcsolatban?

- a) Holtpontmentes lesz a rendszer, ha alkalmazzuk az erőforrásokra vonatkozó teljes sorrendezés algoritmust.
- b) A holtpont szükséges feltétele a hurok az erőforrás-foglalási gráfban.

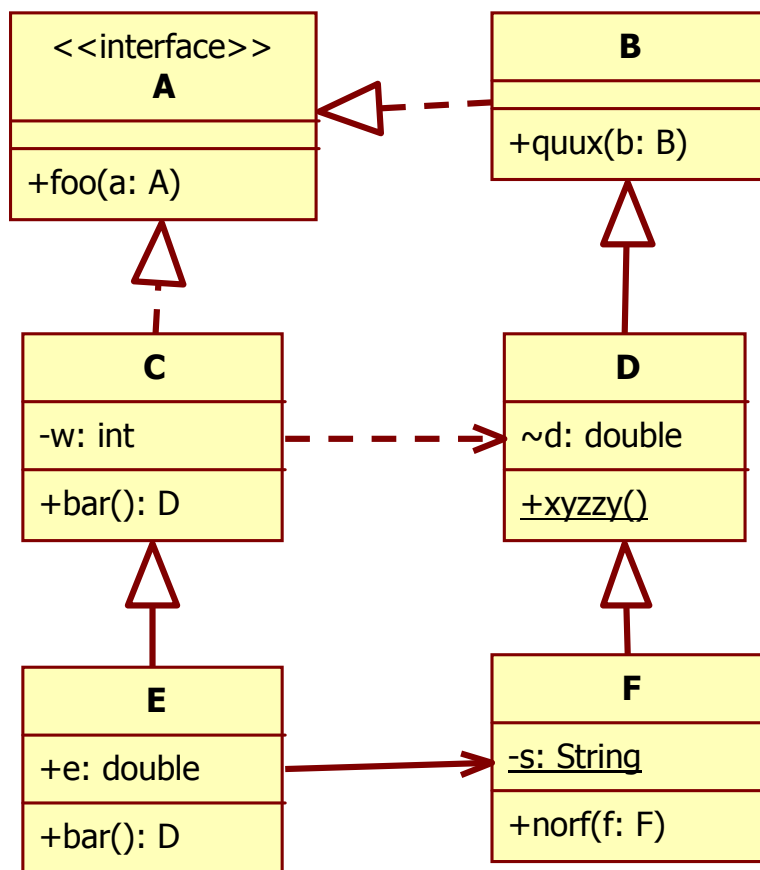
Megoldás: b) (mert az irányított hurok szükséges feltétel.

Az a) helyes, mert ha alkalmazhatjuk (erre nem tér ki a kérdés), akkor az erőforrásokra vonatkozó teljes sorrendezés algoritmus megvalósításával a rendszerben nem lehet holtpont, hiszen nem alakulhat ki körkörös várakozás az erőforrásokra.)

pont(0,5):

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

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



- 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 (- -)

D `foo(a:A)` függvénye módosíthatja egy paraméterül kapott F objektum `d` attribútumát, mert a `d` protected.

Megoldás: **E**

pont(1):

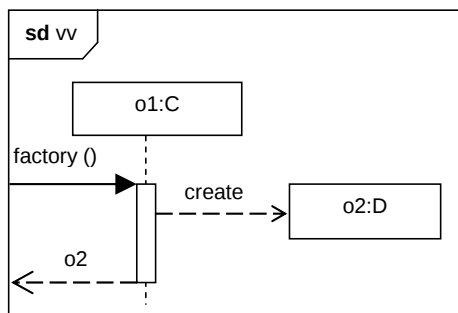
2. A fenti ábra alapján a D osztály megvalósítja-e a `quux(b:B)` szignatúrájú metódust?

- A – igen, mert az osztály nem absztrakt
 B – nem, mert az osztályban nincs ilyen metódus feltüntetve
 C – nem lehet eldönteni, mert a metódus a D-ben lehet absztrakt is

Megoldás: **A**

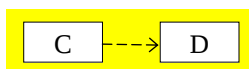
pont(1):

3. Adott az alábbi szekvenciadiagram. Milyen kapcsolat olvasható ki belőle a C és D osztály között? Egyszerű osztálydiagramon jelölje is!



- A – dependencia
B – asszociáció
C – aggregáció
D – kompozíció

Megoldás: A



pont(1):

4. A CMM melyik szintjeire nem igaz az alábbi állítás?

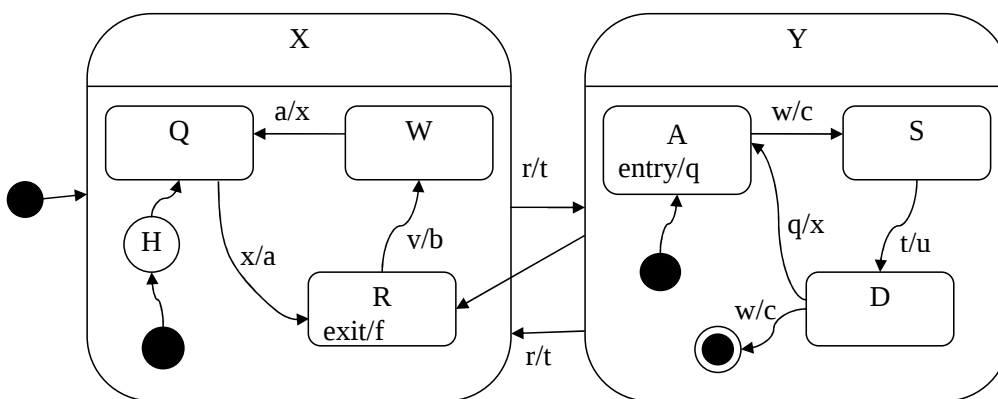
A szoftverfejlesztési folyamat tevékenységei dokumentáltak, szabványosítva vannak, és a szervezet szabványos szoftverfejlesztési folyamatává integrálták őket.

1 2 3 4 5

Megoldás: 1, 2

pont(1):

5. Adott az alábbi UML2 állapotgép (state chart).



Válassza ki az alábbiak közül a kezdés után az r, w, t, w eseményszekvencia hatására kialakuló végállapotot!

A S D Q W R

Megoldás: R

pont(1):

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

1. Adja meg két-három pontban, miben és hogyan segítenek a tervezési minták a szoftvertervezés során! Figyelem: NE a tervezési minta definícióját adja meg!

Megoldás: Növelik a rendszer karbantarthatóságát, módosíthatóságát;
növelik az egyes részek újrafelhasználhatóságát;
segítenek megtalálni a nem maguktól értetődő osztályokat.

pont(1):

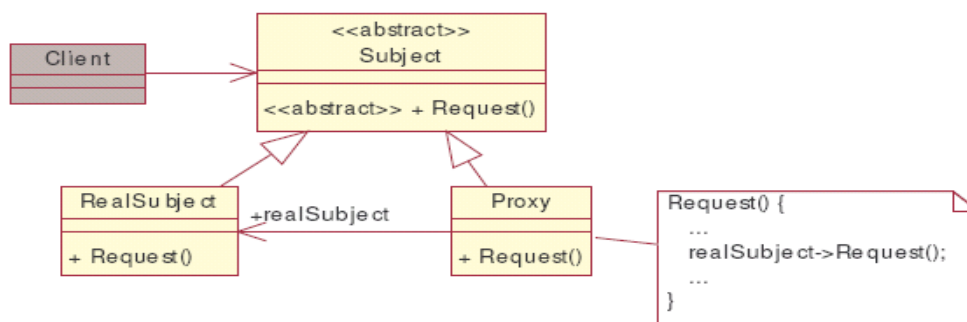
2. Milyen általános problémát old meg az Observer (Megfigyelő) tervezési minta?

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

pont(1):

3. Mutassa be általánosságban, vagy egy példán keresztül a Proxy minta működését, ezen belül rajzolja fel a minta osztálydiagramját!

Megoldás:
Osztálydiagram:



A kliens a *Subject* őss osztályon/interfészen keresztül hivatkozik egy *Proxy* objektumra, ami bizonyos kéréseket maga szolgál ki, de az esetek többségében tovább (is) hív az általa hivatkozott *RealSubject* objektumba.

pont(1):

-
4. Jellemezze röviden a Proxy osztálydiagramon szereplő osztályokat!

Megoldás:

RealSubject: Az eredeti objektumok osztálya (amelyhez szabályozott hozzáférésre van szükség).

Proxy: A helyettesítő objektum osztálya.

Subject: Közös interfészt biztosít a Proxy és a RealSubject számára, így lesz a kliens számára transzparens a helyettesítés.

Client: Számára szükséges a szabályozott módon való hozzáférés biztosítása.

pont(1):

-
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(1):

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

1. Minimalizálja az $F = \{A \rightarrow BA, CD \rightarrow EAF, C \rightarrow ABC, D \rightarrow DF\}$ függéshalmazt!

Megoldás: $F_{min} = \{A \rightarrow B, C \rightarrow A, D \rightarrow F, CD \rightarrow E\}$

pont(1):

2. Az $R(A, B, C, D)$ séma melyik attribútumaiban valósulhat meg redundáns adattárolás funkcionális függés következtében, ha az attribútumok között az $ABD \rightarrow CD$ és $C \rightarrow CD$ függések ismertek?

Megoldás: A D attribútumban.

pont(1):

3. Egy relációt, ill. sémáját az alábbi paraméterek jellemeznék:

Attribútum neve	Kulcs?	Kardinalitás	Típusa	Hossza (byte)
A1	Igen	900	INTEGER	6
A2	Nem	3.000	DATE	4
A3	Nem	100	CHAR	30

Miért nem létezhet ilyen reláció a valóságban?

Megoldás: Ha A1 kulcs, és csak 900 különböző értéket tud felvenni, akkor ennél nem lehet több rekord a relációban, tehát egyetlen attribútum sem vehet fel ennél több különböző értéket.

pont(1):

4. Egy táblában 100.000 db 100 byte-os rekord található, amelynek a kulcsán B*-fa indexet definiáltak. A kulcs 20 byte-os, egy mutató 5 byte, a blokkméret 1000 byte. Számítsa ki, hogy hány százalékkal nő a kulcs alapján történő keresés ideje abban az esetben, ha minden (adat- és index)blokk csak félig van tele ahhoz képest, amikor minden blokk tele van!

Megoldás: 25%-kal.

pont(1):

5. Adott egy R relációs séma, és az attribútumainak X és Y részhalmazai. Melyek a legtágabb feltételei annak, hogy az $(XY, R \setminus Y)$ sémafelbontás az $X \rightarrow Y$ funkcionális függés esetén veszteségmentes legyen?

Megoldás: Ne legyen Y -nak olyan attribútuma, amely triviálisan függ X -től.

pont(1):

