

<b>MI</b>	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 Mérnökinformatikus szak BME Villamosmérnöki és Informatikai Kar

**2016. január 4.**  
**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.

<b>Főspecializáció</b>	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.

<b>Mellékspecializáció</b>	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)	

<b>AL</b>	Név, felvételi azonosító, Neptun-kód:	pont(15):
	MEGOLDÁS	

1. Legyen  $f(n) = 2016 \cdot \sqrt{n} + 3 \cdot \log_2(n!)$ . Következik-e ebből, hogy

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

igen — nem

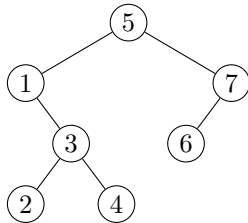
(ii)  $f(n) = O(n^3)$

igen — nem

pont(1):

2. Lehetséges-e, hogy amikor egy 7 csúcsú bináris keresőfában tárolt számokat preorder sorrendben kiolvassuk, akkor az 5, 1, 3, 2, 4, 7, 6 számsorozatot kapjuk? Ha igen, adjon meg egy lehetséges keresőfát!

Megoldás:



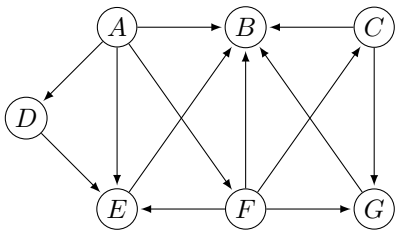
pont(1):

3. A  $V = \{1, 2, 3, 4, 5\}$  csúcsalmazon levő teljes gráf minden élre a piros, kék, zöld színek egyikére van festve. Hányféle lehetséges kifestés van, ha tudjuk, hogy pontosan egy él zöld? Nem szükséges kiszámolni, elegendő egy formulát megadni. (Az a kifestés, amikor az egyes és kettes csúcsokat összekötő él zöld és a többi piros, különbözik például attól, amikor az egyes és hármas él a zöld és a többi piros.)

Megoldás:  $10 \cdot 2^9$

pont(2):

4. Az alábbi gráfon a mélységi bejárást a  $D$  csúcsból indulva úgy hajtjuk végre, hogy amikor több lehetőség van, akkor mindig az ábécésorrend szerinti elsőt választjuk. Adja meg a bejáráás során megkapott mélységi és befejezési számokat, továbbá a gráf egy topologikus sorrendjét (ha van ilyen)!



Megoldás:

	A	B	C	D	E	F	G
mélységi szám	4	3	6	1	2	5	7
befejezési szám	7	1	5	3	2	6	4

Topologikus sorrend:  $A, F, C, G, D, E, B$  (a bef. számokból, de másként is lehet: az  $A, F, C, G, B$  sorrend fix,  $D$  az  $A$  és  $G$  között bárhol, az  $E$  a  $D$  után, de  $B$  előtt)

pont(2):

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

Minden  $k \geq 1$  esetén, ha  $v_1, v_2, v_3, \dots, v_{2k+1} \in V$  csupa különböző csúcs, akkor vagy  $\{v_1, v_{2k+1}\} \notin E$  vagy van olyan  $1 \leq i \leq 2k$ , hogy  $\{v_i, v_{i+1}\} \notin E$ .

Fogalmazza meg, milyen ismert gráftulajdonságot ír le  $\mathcal{T}$ !

Megoldás:  $G$  egy páros gráf (nincs benne páratlan kör).

pont(2):

6. Sok barátunk van, de sajnos közülük nem mindenki szereti egymást. A kapcsolatok milyenségét egy 5 fokú skálán osztályozzuk, 5 jelenti, hogy jóban vannak, 1 pedig, hogy ki nem állhatják egymást. Minden lehetséges párt egy ilyen osztályzattal jellemzünk (ami tehát a kölcsönös viszonyt jellemzi). A kínos helyzetek elkerülésére szilveszterre úgy szeretnénk minél több barátunkat meghívni, hogy ne legyen közöttük hármasnál rosszabb kapcsolat.

Melyik ismert gráfelméleti feladattal írható le a probléma és mi a kérdéses gráf? (Nem kell megoldást adni a problémára, csak át kell fordítani a meghívottak meghatározását egy ismert problémára.)

*Megoldás:* A  $G(V, E)$  gráf csúcsai a barátok, élei a legalább 3-as párok. Ebben kell maximális (minket is tartalmazó) klikket találni.

pont(2):

7. Tekintsük a következő eldöntési problémákat!

Jelöljön  $G$  egy egyszerű irányítatlan gráfot, melynek élei pozitív számokkal súlyozottak.

$\mathcal{A}$ : Adott a  $G$  gráf. Van  $G$ -nek Hamilton-köre?

$\mathcal{B}$ : Adott a  $G$  gráf és egy  $k > 0$  egész szám. El lehet-e jutni  $G$ -ben bármely csúcsból bármely csúcsba egy legfeljebb  $k$  súlyú úttal?

$\mathcal{C}$ : Adott a  $G$  gráf. El lehet-e jutni  $G$ -ben bármely csúcsból bármely csúcsba egy legfeljebb 500 súlyú úttal?

Tegyük fel, hogy  $P \neq NP$ . Lehetséges-e hogy

$\mathcal{A}$  Karp-redukálható (polinomiálisan visszavezethető)  $\mathcal{B}$ -re. igen – nem

$\mathcal{A}$  Karp-redukálható (polinomiálisan visszavezethető)  $\mathcal{C}$ -re. igen – nem

$\mathcal{C}$  Karp-redukálható (polinomiálisan visszavezethető)  $\mathcal{A}$ -ra. igen – nem

pont(2):

8. Egy zárthelyi eredménye minden hallgatónál egy pontszám. Vázzon olyan adatszerkezetet, amivel az alábbi műveletek megvalósíthatók:

BESZÚR(kód): az adott NEPTUN-kódú hallgatót beveszi a zárthelyit írók közé, az elért pontszámát 0-ra állítja be;

NÖVEL(kód,  $m$ ): az adott NEPTUN-kódú hallgató tárolt pontszámához hozzáad  $m > 0$  pontot;

MENNYI(kód): megadja, hogy az adott kódú hallgatónak hány pontja van;

LEGJOBB: a legjobb zárthelyit írók NEPTUN-kódját kiírja, azaz azokét a hallgatókét, akik a legmagasabb pontszámot érték el.

A LEGJOBB művelet lépésszáma a kiírandó hallgatók számával legyen arányos, a másik három műveleté pedig  $O(\log n)$ , ahol  $n$  az összes hallgató száma.

*Megoldás:*

A (kód, pontszám) párokat tároljuk egy piros-fekete fában (vagy 2-3 fában vagy AVL fában) a kód szerint, és külön nyilvántartjuk az aktuális legnagyobb pontszámot (max), mutatókkal a megfelelő fabeli csúcsokra. (BESZÚR a szokásos, MENNYI egy KERES-t használ. NÖVEL-nél figyelni kell, elérjük vagy túlhaladjuk-e a max értékét, az első esetben a max-hoz tartozó mutatókhoz ezt is hozzávesszük a másodikban az eddigieket helyettesítjük vele. A LEGJOBB a mutatók alapján kiírja a kódokat.)

pont(3):

<b>H</b>	Név, felvételi azonosító, Neptun-kód:  MEGOLDÁS	pont(7,5):
----------	---	------------

1. Az alábbiak közül mely állítás(ok) igaz(ak) az SMTP-re?

- a) A POP3/IMAP4 kliensek ennek segítségével fogadhatnak levelet.
  - b) Az MTA-k egymásnak ezen protokoll segítségével továbbítják az e-maileket.
  - c) Használata esetén nem szükséges megadni a küldő e-mail címét.
  - d) A többi válasz közül egyik sem helyes.
- Megoldás: **b)**

pont(1):

2. Az alábbi routing protokollok közül melyik szolgál autonóm rendszerek közötti útvonalirányításra?

- a) IS-IS (Intermediate System to Intermediate System)
  - b) OSPF (Open Shortest Path First)
  - c) RIP (Routing Information Protocol)
  - d) EIGRP (Enhanced Interior Gateway Routing Protocol)
  - e) BGP (Border Gateway Protocol)
- Megoldás: **e)**

pont(1):

3. A Bellman–Ford-algoritmus alkalmazása során egy adott időpontban a hálózat *A* csomópontja a következő állapotvektort tartja nyilván:

<i>B</i> , 1	<i>C</i> , 2	<i>D</i> , 3	<i>E</i> , 4	<i>F</i> , 1
--------------	--------------	--------------	--------------	--------------

Megérkezik *B*-től a következő állapotvektor:

<i>A</i> , 1	<i>C</i> , 3	<i>D</i> , 2	<i>E</i> , 2	<i>F</i> , 3
--------------	--------------	--------------	--------------	--------------

Mely bejegyzéssel/bejegyzésekkel bővíti ill. módosítja *A* az állapotvektorát?

Megoldás: *E*, 3

pont(1):

4. Mi váltotta fel az osztályalapú címzést IPv4-ben? Az új címzés nevét várjuk válaszként.

Megoldás: CIDR (Classes Inter-Domain Routing, osztálymentes címzés)

pont(1):

5. Egészítse ki az alábbi mondatot:

A ..... egy olyan TCP állapotváltozó, ami az adó által küldhető adatmennyiséget korlátozza le egy érték alá, hogy elkerüljük a torlódást a hálózatban.

Megoldás: torlódási ablak

pont(1):

6. Amennyiben ismerjük az adott eszköz IP-címét és a hozzá tartozó adatkapcsolati rétegbeli címre van szükség, akkor az Address Resolution Protocolt használjuk IPv4 esetén. A folyamat elején milyen kérést küldünk az alhálózaton belül?

Megoldás: broadcast

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?

Megoldás: 420 byte

pont(1,5):



<b>O</b>	Név, felvételi azonosító, Neptun-kód:  MEGOLDÁS	pont(7,5):
----------	---	------------

*Figyelem! Minden feladatnál csak egy helyes válasz van!*

1. Az alábbi állítások közül melyik *igaz* a multiprogramozott rendszerekkel kapcsolatban?

- a) A multiprogramozott rendszerekben a lassú perifériák korlátozzák a rendszer működését, hiszen a CPU-nak meg kell várnia a perifériás műveletek befejezését.
- b) A 100%-os CPU-kihasználtság a gyakorlatban is elérhető multiprogramozott rendszerekben.
- c) A multiprogramozott rendszerekben egy feladat addig fut, amíg az várakozni nem kényszerül a CPU-ra.
- d) A multiprogramozott rendszerek FIFO ütemezőt használnak.

*Megoldás: c)* ( a) – pont ezt a problémát küszöbölik ki a multiprogramozott rendszerek. b) – 100% CPU kihasználtság az adminisztrációs feladatok miatt még elméletileg sem érhető el, nem hogy gyakorlatilag. d) – a multiprogramozott működés nem mond semmit az ütemező algoritmusról. )

pont(1):

2. Az alábbi állítások közül melyik *hamis* az operációs rendszerek rétegszerkezetével kapcsolatban?

- a) Az operációs rendszerekben csak az eszközmeghajtók tartalmaznak hardverspecifikus kódot.
- b) Az operációs rendszer fontos feladata a memóriakezelés.
- c) Az operációs rendszer magja (kernel) többnyire platformfüggetlen módon kerül megvalósításra.
- d) A rendszerhívási felület az operációs rendszer egyik kernel módban futó része.

*Megoldás: a)* (a kernel is HW specifikus, ezért alkalmazzák a HAL-t a kernel és a HW platformfüggetlen kapcsolatának megteremtésére. A b) és a d) definíciószerűen igaz. A c) is igaz, éppen azért, mert az a) hamis. )

pont(1):

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

- a) A FIFO algoritmus alkalmazása esetén megfigyelhető a konvoj hatás.
- b) Az SJF algoritmus esetén nem jelentkezik a konvoj hatás.
- c) Az RR algoritmus esetén a konvoj hatás nem jelentkezik az algoritmus preemptív volta miatt.
- d) Az SRTF esetén nem jelentkezik a konvoj hatás.

*Megoldás: b)* (az SJF nem preemptív, azért egy kicsit korábban érkező nagy löketidejű feladat fel tudja tartani a később érkező kis löketidejű feladatokat.)

pont(1):

4. Az alábbi állítások közül melyik *hamis* a feladatok tipikus állapotátmeneti diagramjával kapcsolatban?

- a) Az erőforrást kérő feladat VÁRAKOZÓ állapotba kerül.
- b) Egy operációs rendszeren belül maximum annyi feladat lehet futó állapotban, ahány végrehajtó egységünk van.
- c) Kooperatív operációs rendszerben a yield() rendszerhívás hatására FUTÓ állapotból FUTÁSRA KÉSZ állapotba kerülhet egy feladat.
- d) Feladat mindig FUTÁSRA KÉSZ állapotban jön létre.

*Megoldás: a)* (csak akkor kerül várakozó állapotba, ha az erőforrás foglalt, ha az erőforrás szabad, akkor megkapja az erőforrást, és fut tovább.)

pont(1):

---

5. Az alábbi állítások közül melyik *hamis* a folyamatra (process), szálakat támogató folyamatalapú rendszerben (pl. Windows vagy UNIX)?

- a) A folyamat egy végrehajtás alatt álló program.
- b) Egy programból több különböző folyamatot létrehozhatunk.
- c) A folyamatok közös memórián keresztül kommunikálnak egymással.
- d) A folyamatok közös erőforrásaira (pl. két folyamat által is megnyitott file) biztosítani kell a kölcsönös kizárást.

*Megoldás: c)* (közös memória alapértelmezett módon nincs, a kommunikáció OS hívásokkal történik (POSIX/SYSV shared memória egy olyan kompromisszum, ami valójában nem sérti meg az alapelvet, mert a memóriamegosztás az OS által felügyelt és részleges.)

pont(1):

---

6. Mely állítás *hamis* a holtponttal kapcsolatban?

- a) A holtpontészlelés és -feloldás nem feltétlenül oldja meg a problémát (pl. livelock lehet az eredménye).
- b) A holtpont többnyire a rendszer feladatainak csak egy csoportjára terjed ki, vagyis a rendszer részben működőképes maradhat.
- c) A holtpont egy versenyhelyzet, amelyben a feladatok egymásra váró állapotba kerülnek.
- d) A holtpont elkerülhető számláló típusú szemaforok használatával.

*Megoldás: d)*

pont(1):

---

7. Az alábbi lapozással és lapcsere algoritmusokkal kapcsolatos kérdések közül melyik *igaz*?

- a) Az előre tekintő lapozás csökkenti a laphiba gyakoriságot.
- b) A legrégebbi lap (FIFO) lapcsere algoritmus a jövőbe tekint.
- c) Az utóbbi időben nem használt (NRU) algoritmus esetén a frissen behozott lapot a tárba kell fagyasztani, mert az könnyen a lapcsere algoritmus áldozatává válhat egyébként.
- d) A legrégebben nem használt (LRU) algoritmus megvalósítása erőforrás-igényes, ezért sokszor annak a közelítéseit alkalmazzuk.

*Megoldás: d)* (Az a) esetén nincs garancia a csökkenésre, akár növelheti is, ha nem jól tekint "előre". A FIFO a múltba tekint, ezért a b) hibás. A c) hibás, mert a LFU algoritmusnál van szükség a tárba fagyasztásra a megadott esetben.)

pont(1):

---

8. Az alábbi két állítás közül melyik *igaz* a láncolt listás tároláson alapuló fájlrendszer esetén?

- a) A fájlokban tárolt adatok szekvenciális és indexelt (közvetlen) elérése is egyszerűen és hatékonyan megoldható.
- b) A fájlokban tárolt adatok egymás utáni, szekvenciális blokkokba rendezése (töredezésmentesítés) javítja az írási és olvasási teljesítményt is.

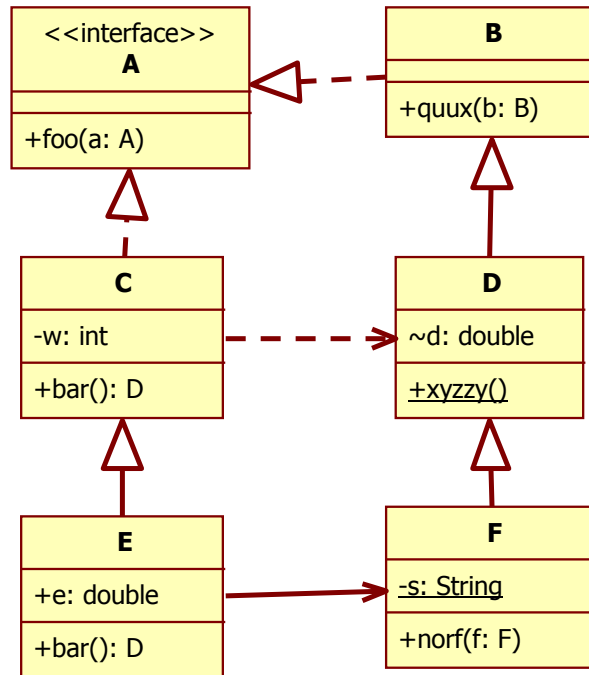
*Megoldás: b)* (az indexelt elérés nagy erőforrás igényű (végig kell menni a láncolt listán).)

pont(0,5):



<b>S1</b>	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 (- -)

F-nek a `foo(a: A)` függvénye módosíthatja egy paraméterül kapott E objektum `e` attribútumát, mert az `e` publikus.

Megoldás: **D**

pont(1):

2. A szoftverfejlesztés melyik fázisának célja „a rendszer fő komponenseinek azonosítása és a közöttük fennálló együttműködés definiálása”?

Megoldás: Architektúrális tervezés

pont(1):

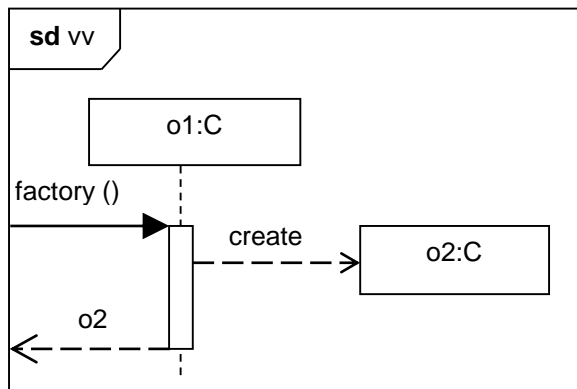
3. Nevezzen meg egy statikus verifikációs technikát!

Megoldás: felülvizsgálat, átvizsgálás, review, audit

pont(1):

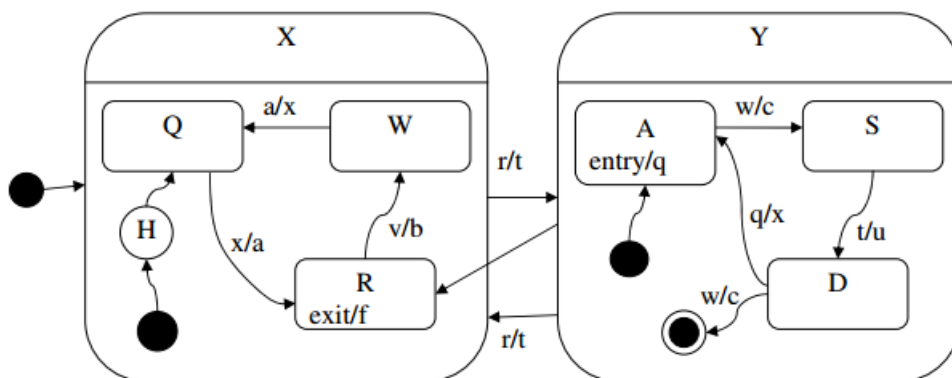
4. Egy C osztályú o1 objektum factory() metódusa elkészíti a C osztály egy új példányát (o2) és az új példányt átadja a metódus meghívójának. Rajzoljon UML2 szekvencia-diagramot!

Megoldás:



pont(1):

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



Jelölje meg, hogy a kezdés után az x, r, w, r eseményszekvencia hatására melyik lesz a kialakuló végállapot!

A S D Q W R

Megoldás: R

pont(1):

<b>S2</b>	Név, felvételi azonosító, Neptun-kód:  MEGOLDÁS	pont(5):
-----------	---	----------

1. Adja meg két-három pontban, miben és hogyan segítenek a tervezési minták a szoftvertervezés során!

*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 a Factory Method (Metódusgyár) tervezési minta?

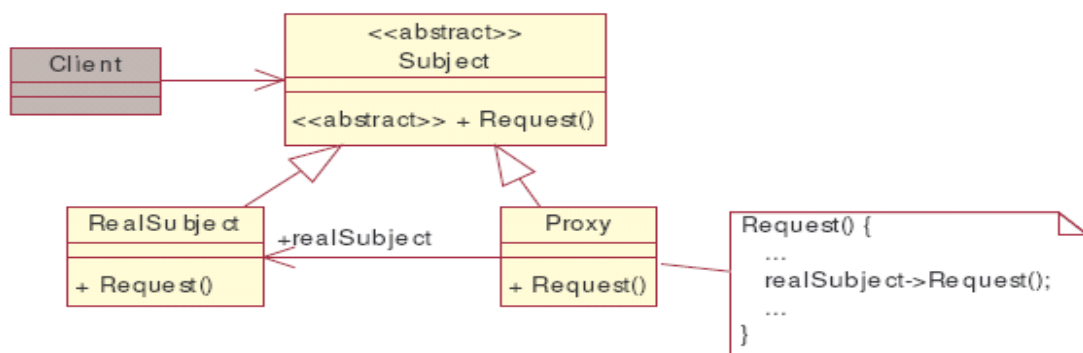
*Megoldás:*

Interfészt definiál az objektum létrehozására, de a leszármaztatott osztályra hagyja annak eldöntését, hogy konkrétan melyik osztályból kell példányt létrehozni. Akkor használjuk, ha egy osztály nem látja előre annak az objektumnak az osztályát, amit létre kell hoznia, valamint ha egy osztály azt szeretné, hogy leszármazottai határozzák meg azt az objektumot, amit létre kell hoznia.

pont(1):

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

*Megoldás:*



A kliens a Subject őszosztályon/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):

---

<b>AD</b>	Név, felvételi azonosító, Neptun-kód:  MEGOLDÁS	pont(5):
-----------	---	----------

1. Hányadik normálformájú az  $R(A, B, C, D, E, F)$  atomi attribútumokból álló relációs séma az alábbi függéshalmaz esetén?

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

Megoldás: 3NF

pont(1):

2. Minimalizálja a  $\{A \rightarrow BC, C \rightarrow DE, E \rightarrow B, ED \rightarrow F, AF \rightarrow H\}$  függéshalmazt!

Megoldás:  $A \rightarrow C, C \rightarrow D, C \rightarrow E, E \rightarrow B, ED \rightarrow F, A \rightarrow H$

pont(1):

3. Adott az  $R(ABCD)$  séma az  $F = \{AB \rightarrow C, C \rightarrow D\}$  függéshalmazzal. Minden attribútum egyszerű. Egyetlen nemtriviális funkcionális függéssel bővítve  $F$ -et érje el, hogy a séma pontosan 1NF legyen!

Megoldás: pl.  $A \rightarrow D$

pont(1):

4. Melyek igazak az alábbi állítások közül?

Van olyan pontosan 3NF séma (nem BCNF), hogy létezik egy rá illeszkedő reláció,

- a) amely az egyik másodlagos attribútumában redundáns.
- b) amelynek egyetlen másodlagos attribútumában sincs redundancia.
- c) amely az egyik elsődleges attribútumában redundáns.

Megoldás: b), c)

pont(1):

5. Egy 1000 rekordból álló állományt ritka index szervezéssel tárolunk. A rekordhossz 850 bájt, egy blokk kapacitása (a fejrész nem számítva) 4000 bájt. A kulcs 50 bájt, egy mutatóhoz 18 bájt kell. Egy blokkművelet ideje 2 ms. Hány blokkot foglal el az adatállomány?

Megoldás: 250 blokkot

pont(1):