

AL	Név, azonosító:	pont(30) :
-----------	-----------------	------------

1. Jelölje $T(n)$ egy algoritmus maximális lépésszámát az n hosszú bemeneteken. Tudjuk, hogy $T(1) = 7$ és $T(n) \leq 3n + T(n - 1)$, ha $n > 1$. Következik-e ebből, hogy
- (i) $T(n) = O(n^{\log n})$?
 - (ii) $T(n) = O(n^2)$?

Megoldás: Mindkettő következik.

pont(2):

2. Egy rendezés során a kezdeti 5, 6, 3, 8, 4, 1, 2, 7 sorrendből valahány lépés után a pillanatnyi helyzet a 3, 5, 6, 8, 1, 2, 4, 7 sorrend. Lehetett-e a rendezés a beszúrásos, az összefésüléses, illetve a gyorsrendezés?

Megoldás: Beszúrásos: nem; Összefésüléses: igen; Gyorsrendezés: nem.

pont(2):

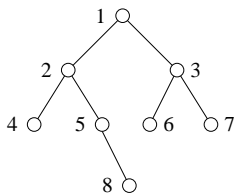
3. Hány különböző 4 hosszú kör található a $K_{n,m}$ teljes páros gráfban?

Megoldás: $\binom{n}{2} \cdot \binom{m}{2}$

pont(2):

4. Egy bináris fa csúcsai az 1 és 8 közötti egész számokkal vannak megcímkézve. A csúcsok preorder bejárás szerinti sorrendje: 1, 2, 4, 5, 8, 3, 6, 7, a postorder bejárás szerinti sorrend pedig 4, 8, 5, 2, 6, 7, 3, 1. Rajzoljon fel egy, a feltételeknek megfelelő fát!

Megoldás:



Megjegyzés: a 8 csúcs az 5-nek baloldali gyereke is lehet.

pont(4):

5. Egy ország városai között csak busszal vagy vonattal tudunk közlekedni. Minden X és Y városra ($X \neq Y$) adott, hogy van-e olyan buszjárat, amivel átszállás nélkül el tudunk jutni X -ből Y -ba. Ha van ilyen járat, akkor adott az is, hogy X és Y között mennyibe kerül egy buszjegy. Hasonlóan adottak a vonattal való elérhetőségek és jegyár-információk is. Egy rögzített A városból szeretnénk egy rögzített B városba eljutni a lehető legolcsóbb módon. Utunk során át is szállhatunk és válthatunk buszról vonatra, vonatról buszra, akár többször is.

A legolcsóbb útvonal árának meghatározására melyik ismert algoritmust használná, és milyen bemeneten futtatná azt?

Megoldás: Dijkstra- vagy Bellman-Ford-algoritmus.

A gráf csúcsai a városok, kettő között akkor van él, ha van közöttük busz vagy vonat járat, súlya a kisebb költség. Az A -ból a B -be keresünk legrövidebb utat.

pont(4):

6. Tegyük fel, hogy $P \neq NP$. Mindkét alábbi feladatra döntse el, hogy P-ben van-e:

A G irányítatlan gráfban létezik

(i) legfeljebb 100 élű út?

(ii) legalább k élű út (ha k része a bemenetnek)?

Megoldás:

(i) P-beli

(ii) nincs P-ben (mert NP-teljes).

pont(4):

7. Az \mathcal{A} halmaz álljon azokból a $G = (A, B, E)$ páros gráfokból, melyekre van olyan $X \subseteq E$, hogy minden $a \in A$ csúcshoz létezik $b \in B$ csúcs, hogy $\{a, b\} \in X$, valamint $\{a_1, b_1\}, \{a_2, b_2\} \in X$ esetén $a_1 = a_2 \Leftrightarrow b_1 = b_2$. Vázzon egy polinom lépésszámú algoritmust, ami eldönti, hogy egy éllistával megadott $G = (A, B, E)$ páros gráf beletartozik-e az \mathcal{A} halmazba!

Megoldás: A magyar módszert használjuk. Ha az ezzel talált párosítás lefedi A -t, akkor G az \mathcal{A} halmazhoz tartozik, egyébként nem.

pont(6):

8. Legyen $s_1 s_2 \dots s_n$ és $t_1 t_2 \dots t_m$ két olyan karaktersorozat, mely nullákból és egyesekből áll. Azt szeretnénk, hogy az $n \times m$ méretű A mátrix $A[i, j]$ eleme tartalmazza azt a legnagyobb k számot, melyre az $s_1 s_2 \dots s_i$ és a $t_1 t_2 \dots t_j$ karaktersorozatok utolsó k tagja megegyezik. Adjon eljárást, ami az A tömböt $O(nm)$ lépésben kitölti.

Egy lehetséges megoldás, ha az alábbi módon és sorrendben töltjük ki a tömb elemeit (dinamikus programozás).

$$A[1, j] = \begin{cases} 0 & \text{ha } s_1 \neq t_j \\ 1 & \text{ha } s_1 = t_j \end{cases} \quad j = 1, \dots, m$$

$$A[i, 1] = \begin{cases} 0 & \text{ha } s_i \neq t_1 \\ 1 & \text{ha } s_i = t_1 \end{cases} \quad i = 1, \dots, n$$

$$A[i, j] = \begin{cases} 0 & \text{ha } s_i \neq t_j \\ 1 + A[i-1, j-1] & \text{ha } s_i = t_j \end{cases} \quad i = 2, \dots, n, \quad j = 2, \dots, m$$

pont(6):

H	Név, azonosító:	pont(15) :
----------	-----------------	------------

1. Az alábbiak közül mely(ek) *nem* a TCP feladata(i)?
- a) Útvonalválasztás
 - b) Forgalm szabályozás
 - c) Váltás a karakterkészletek között
 - d) Sorrendhelyes átvitel
 - e) A fentiek közül mindegyik a TCP feladata.

Megoldás: **a), c)**

pont(2):

2. Melyik megfogalmazás írja le legjobban a CIDR (Classless Inter-Domain Routing) lényegét az alábbiak közül?
- a) Alhálózatokra osztjuk az A- vagy a B-osztályú tartományt.
 - b) A címtartományt bárhol meg lehet osztani hálózati és végpont-címre.
 - c) A címtartomány egységes, nem oszlik hálózati és végpontcímre.
 - d) A beérkező IP cím osztályától függetlenül a domainen belül bármilyen privát címet kioszthatunk.
 - e) A fentiek közül egyik válasz sem helyes.

Megoldás: **b)**

pont(2):

3. Az alábbiak közül mely ISO OSI réteg(ek) *nem* található(k) meg a TCP/IP modellben?
- a) Fizikai réteg
 - b) Alkalmazási réteg
 - c) Adatkapcsolati réteg
 - d) Megjelenítési réteg.
 - e) Hálózati réteg
 - f) Viszonyréteg

Megoldás: **d), f)**

pont(2):

4. Az alábbiak közül mely (al)réteg(ek)et tartalmazza az IEEE 802.3 szabvány szerinti LAN-architektúra?
- a) Hálózati réteg
 - b) Közeghozzáférési alréteg
 - c) Fizikai alréteg
 - d) Menedzsment réteg
 - e) A fentiek közül egyiket sem tartalmazza

Megoldás: **b), c)**

pont(2):

5. Válassza ki az Ethernet keretek hosszára vonatkozó helyes állítás(oka)t!
- a) Minimumát csak a cím hossza határozza meg.
 - b) Nincs maximális hossz előírva.
 - c) A minimális hosszúságot a kábelhossz és az átviteli sebesség alapján határozták meg.
 - d) Kapcsolt Ethernetnél nem számít a hosszúság.
 - e) A fentiek közül egyik állítás sem helyes.

Megoldás: **c)**

pont(2):

6. Milyen szolgáltatás(oka)t nyújt az UDP?
- a) Sorrendhelyes átvitel
 - b) Portkezelés
 - c) Torlódásvezérlés
 - d) Hibavédő-kódolás a teljes UDP PDU-ra
 - e) A fentiek közül az UDP egyik szolgáltatást sem nyújtja.

Megoldás: **b)**

pont(2):

7. Egészítse ki az alábbi szöveget egy-egy helyes szóval, rövidítéssel!

„A névfeloldás az Interneten a hálózati csomópont neve és annak-címe között teremt egyértelmű kapcsolatot. Ennek megvalósítását egy hierarchikus név- és címrendszer végzi, melyre rövidítéssel hivatkozunk. Ennek a hierarchikus rendszernek a csúcsát képezik a rootnak nevezett névfeloldásért felelős kiszolgálók. A terhelés csökkenthető, ha egy zónát több szerver is kiszolgál, vagy az úgynevezett használatával. Ez utóbbi eljárás azt jelenti, hogy ha a végpont – vagy a nevében eljáró – felold egy nevet, akkor ennek az eredménye tárolásra kerül az ismételt kérések kiszolgálásának felgyorsítása végett.”

Megoldás: IP; DNS; cache (gyorsítótár)

pont(3):

O	Név, azonosító:	pont(15):
----------	-----------------	-----------

1. Adja meg, hogy az alábbi válaszok közül melyek *igazak* illetve *hamisak* !

(i) Szorosan csatolt (elosztott) rendszerekben az egyes gépeknek nincs külön memóriája.

igaz – hamis

(ii) Egy, csak egy folyamat által, kizárólagosan használt file megnyitása felhasználói módban történik.

igaz – hamis

(iii) Lapszervezésű rendszerekben nem lép fel külső tördelődés.

igaz – hamis

(iv) A digitális aláírás két célja: a küldő személy és a küldött szöveg hitelesítése.

igaz – hamis

(v) Egy folyamat munkahalmazának nevezzük a folyamat által egy adott időintervallum alatt használt lapok összességét.

igaz – hamis

(vi) A „(bináris) szemafor” univerzális szinkronizációs eszköz, amely a „kölcönös kizárás”, a „haladás”, és a „véges várakozási idő” feltételét is biztosítja.

igaz – hamis

pont(6):

2. Adja meg a helyes választ az alábbi kérdésekre!

(i) Az éppen futó taszkot megszakítja egy külső interrupt (IT). Preemptív operációs rendszer esetén mindig a megszakított taszk fogja-e visszakapni a futási jogot az IT rutin végrehajtása után?

igen – nem

(ii) Ha folyamatok együttműködése közös memórián keresztül valósul meg, akkor a PRAM modell szerint működő memória önmagában biztosítja-e a helyes működést?

igen – nem

(iii) Igaz-e, hogy a Bélády-féle anomália csak FIFO virtuális lapcsere stratégia alkalmazása esetén következhet be?

igen – nem

pont(3):

3. Válassza ki, melyek azok a tulajdonságok, amelyek *egyaránt* jellemzőek a rövid-, közép- és hosszútávú ütemezőre az alábbiak közül:

a) Valamilyen szempont szerint optimalizálja (javítja) a rendszer működését.

b) Biztosítja a CPU 100%-os kihasználtságát.

c) A kernel része.

d) A folyamatok erőforrás-használatát szabályozza valamilyen célfüggvény szerint.

e) Minden rendszerben megtalálható.

Megoldás: **a), d)**

pont(2):

4. Egy operációs rendszer a Shortest Remaining Time First (SRTF, legrövidebb hátralevő lőketidejű) rövidtávú ütemezési stratégiát alkalmazza. Az ütemezési sorba a következő folyamatok érkeznek.

	Érkezési idő	Lőketidő
P1	0	5
P2	1	3
P3	3	5

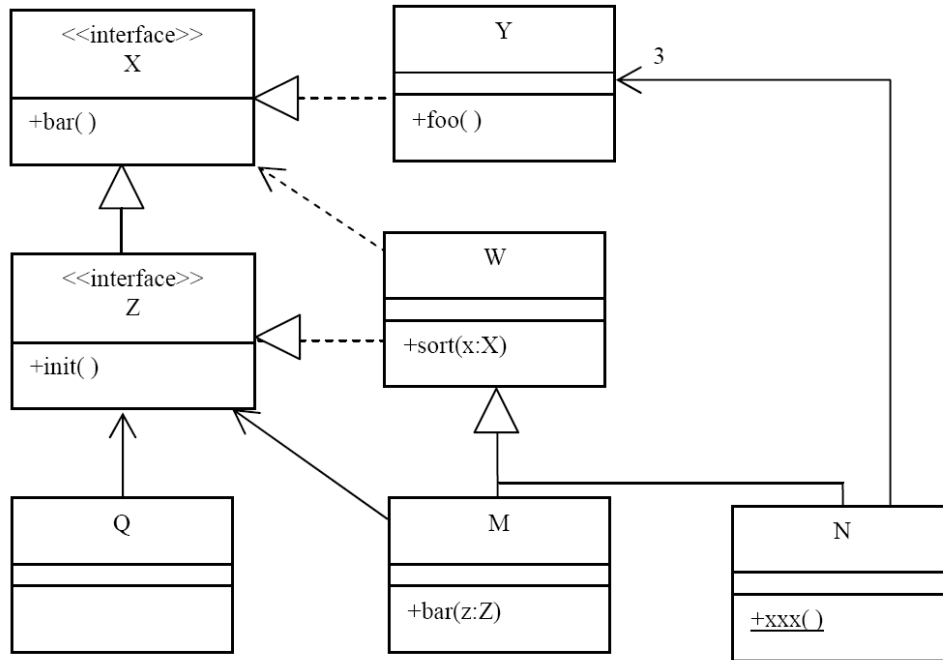
- (i) Melyik folyamat fog futni az 5.-6. óraütés között?
(ii) Mennyi lesz a P3 folyamat várakozási ideje?

Megoldás: (i) P1; (ii) 5

pont(4):

S1	Név, azonosító:	pont(10) :
-----------	-----------------	------------

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



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

(i) W nem helyettesíthető M-mel, mert W-nek nincs bar(z:Z) szignatúrájú metódusa.

(ii) W sort(x:X) metódusa meghívhatja egy paraméterül kapott Y objektum foo() metódusát, mert Y-megvalósítja az X interfészt.

Megoldás: D; D

pont(2):

2. Milyen elemek lehetnek egy adatfolyamábrán, beleértve a környezeti (context) diagramot is?

terminátor

adatfolyam

procesz

adattár

pont(2):

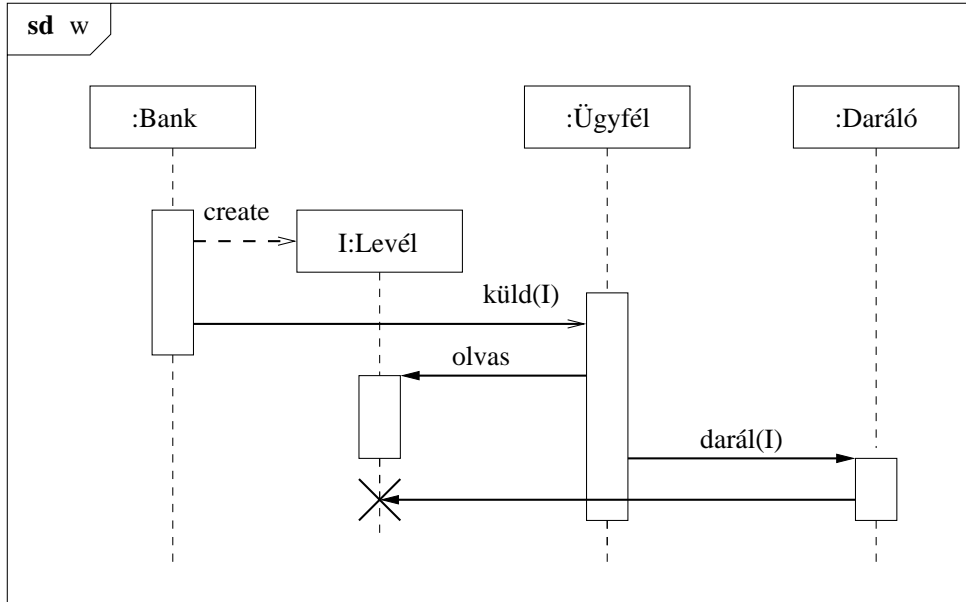
3. Mely fázis előzi meg és követi a szoftverspecifikálást ?

Előző: *Követelmény*

Következő: *Tervezés*

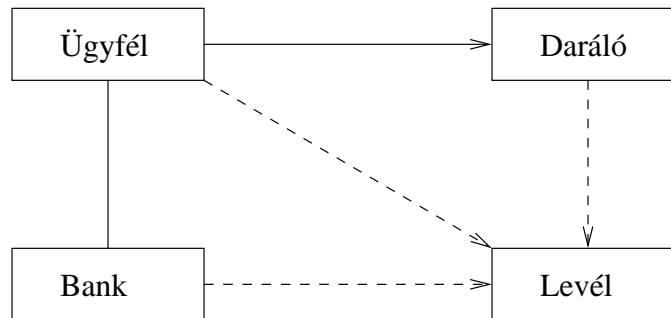
pont(2):

4. Az InterCredit Bank felszólító levelet ír, amelyet elküld egyik ügyfelének, Gerzsonnak, akinek hiteltartozása van. Gerzson a levelet elolvassa, majd a darálón ledarálja. Rajzoljon UML2 szekvenciadiagramot!



pont(2):

5. Tételezze fel, hogy a fenti történetben szereplő objektumok osztályai között nincs más egyéb – a történetből nem kiolvasható – kapcsolat (pl. öröklés)! Az alábbi UML2 osztálydiagramba rajzolja be az osztályok közötti kapcsolatokat!



pont(2):

S2	Név, azonosító:	pont(10):
-----------	-----------------	-----------

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, mmódosíthatóságát;
- növelik az egyes részek újrafelhasználhatóságát;
- segítenek megtalálni a nem maguktól értetődő osztályokat.

pont(2):

2. Milyen általános problémát old meg a Singleton (Egyke) tervezési minta?

Megoldás: Kikényszeríti, hogy egy adott osztályból csak egyetlen objektumot lehessen létrehozni, és ehhez globális hozzáférést biztosít.

pont(2):

3. Mutasson egy C++, Java vagy C# megoldást (kódrészletet) a Singleton tervezési minta implementálására, és mutasson példát a mintának megfelelő osztály használatára!

Egyszerű C++ nyelvű megoldás:

```
class Singleton {
public:
    static Singleton* Instance()
    {
        if (_instance == 0) {
            _instance = new Singleton;
        }
        return _instance;
    }
    void Print() { ... }
protected:
    Singleton() {}
private:
    static Singleton* _instance;
};
Singleton* Singleton::_instance = NULL;
```

Példa használatra:

```
int main()
{
    Singleton::Instance() ->Print();
    ...
}
```

pont(2):

4. Jellemezze az előző pontban megadott megoldást, adja meg a megoldás kulcsgondolatait!

Megoldás: A Singleton osztály az `_instance` statikus tagváltozóban tárolja az egyetlen példányra mutató pointert. Ennek kezdeti értéke `NULL`. Az egyetlen példányhoz hozzáférni az `Instance` statikus tagfüggvénnyel lehet. Első híváskor ez létrehozza az új objektumot, és eltárolja az `_instance` tagban. A későbbi hívások során már ezzel tér vissza. Az osztály konstruktora védett (`protected`), így garantált az, hogy kívülről, az `Instance` tagfüggvény megkerülésével ne lehessen további példányokat létrehozni. Az egyetlen példányhoz a globális hozzáférést az `Instance` statikus tagfüggvény biztosítja.

pont(2):

5. Adja meg röviden a webalkalmazásokra vonatkozóan a szerveroldali szkript fogalmát!

Megoldás: A webszerver által futtatott szkript, melynek elsődleges feladata a kéréssel érkező felhasználói input feldolgozása, és ennek függvényében a kimeneti HTML oldal renderelése.

pont(2):

AD	Név, azonosító:	pont(10):
-----------	-----------------	-----------

1. Végezzen relációanalízist az alábbi P–Q állításpárok között! P és Q önmagában is lehet igaz vagy hamis, továbbá az is eldöntendő, hogy van-e logikai kapcsolat közöttük. Ennek megfelelően a lehetséges válaszok:

- A** – P igaz, Q igaz és van összefüggés
B – P igaz, Q igaz, de nem kapcsolódnak
C – P igaz, Q hamis
D – P hamis, Q igaz
E – mindkettő hamis

P1: Egy tranzitív funkcionális függés lehet triviális függés is,
mivel

Q1: egy attribútumhalmaztól triviálisan függ a részhalmazának részhalmaza.

P2: Ha egy legalább 1NF relációs séma minden kulcsa egyetlen attribútumból áll, akkor a séma automatikusan legalább 2NF is,

ezért

Q2: bármely pontosan 1NF sémát veszteségmentesen és függőségörzően fel lehet bontani legalább 2NF részsémákba.

P3: Egy elsődleges attribútum lehet kulcs, de nem feltétlenül elsődleges kulcs mert

Q3: az elsődleges kulcs egy attribútum, ami mindig elsődleges attribútum.

Megoldás: **B, B, C**

pont(6):

2. Többszörös választás (karikázza be valamennyi helyeset)!

Az alábbiak közül melyiket zárják ki a BCNF sémák?

- a)** függőségörzés (nemtriviális függéseket tartalmazó függéshalmaz esetén)
b) értékfüggő kényszerek érvényesítése
c) funkcionális függés alapú redundancia
d) nemtriviális funkcionális függés
e) összetett (nem atomi) attribútum
f) több kulcs egy sémában
g) másodlagos attribútum
h) ismétlődő attribútumérték

Megoldás: **c), e)**

pont(2):

3. Adja meg az $F = \{A \rightarrow B, AB \rightarrow CD, B \rightarrow AC, CD \rightarrow E, D \rightarrow A\}$ funkcionális függéshalmaz egy minimális fedését!

Megoldás: $F_{min} = \{A \rightarrow B, A \rightarrow D \text{ (vagy } B \rightarrow D), A \rightarrow C \text{ (vagy } B \rightarrow C), B \rightarrow A, D \rightarrow E, D \rightarrow A\}$

pont(2):

