

ÉRETTSÉGI VIZSGA • 2010. október 18.

**INFORMATIKAI
ALAPISMERETEK**

**EMELT SZINTŰ ÍRÁSBELI
ÉRETTSÉGI VIZSGA**

**JAVÍTÁSI-ÉRTÉKELÉSI
ÚTMUTATÓ**

**NEMZETI ERŐFORRÁS
MINISZTERIUM**

Fontos tudnivalók

I. rész

Általános megjegyzések:

- Ha nem a kérdésben meghatározottak szerint válaszol, akkor a válasz nem fogadható el! (Pl.: **H** betű helyett nem válaszolhat **N** betűvel.)
- Ha egy kérdésre a jó válasz(ok) mellett a vizsgázó válaszában hibás választ is megjelöl, akkor a kérdésre adható pontszámból le kell vonni a rossz válaszok számát. Negatív pontszám nem adható, ezért több hibás válasz esetén a minimális pontszám nullánál kevesebb nem lehet.
Pl.: Ha egy jó válasz mellett a vizsgázó egy hibás választ is bejelöl, akkor 0 pontot kell adni. Ez nem vonatkozik azokra a kérdésekre, ahol a (**minden helyes részválasz 1 pont**) szöveg szerepel.
- A kifejtős kérdések (nem feleletválasztós) válaszainál nem a szó szerinti, hanem a helyes tartalmi, illetve a lényegi válaszok megadását kell értékelni. Ha a vizsgázó válaszában a tartalmi vonatkozásai megfelelnek a megoldási útmutatóban megadott válasznak, akkor a válasza adható pontot meg kell adni. Ha csak kis részben, vagy pedig nem felel meg a kapott válasz, akkor pont nem jár a válaszáért.
- A pontszámok az **I.** részben a megadott részletezésnél tovább nem bonthatók (0,5 pont nem adható)
- Egyes esetekben előfordulhat, hogy egy általánostól eltérő rendszer használata miatt valamely kérdésre a vizsgázó nem a várt választ adja, de *a válasza és az indoklása elfogadható*. Ilyen esetben a kérdésre adható pontszámot meg kell adni.
Pl.: Táblázatkezelőkben magyar beállításnál a tizedesek elválasztásának a jele a **vessző**, és ez a várt válasz. Ha a vizsgázók munkájuk során angol beállítást használnak, vagy a vizsgázó odaírja ezt megjegyzésként, akkor az előző helyett az angol beállítású környezetben használt **pont** lesz a helyes válasz.

A javítási-értékelési útmutatóban feltüntetett válaszokra kizárólag a megadott pontszámok adhatók.

A megadott pontszámok további bontása csak ott lehetséges, ahol erre külön utalás van. Az így kialakult pontszámok csak egész pontok lehetnek.

Hardver

- 1) b..... 1 pont
- 2) d..... 1 pont
- 3) Összesen: 3 pont
 A hibajavító kódok segítségével a bithibák javíthatók, ha az egyszerre meghibásodott bitek száma kevesebb, egy előre rögzített értéknél. 1 pont
 Ha a hibás bitek száma eléri ezt az értéket, akkor a hibát csak jelezni tudja, de javítani nem..... 1 pont
 A hibajavító kódok használata biztonságosabbá, de egyúttal drágábbá is teszi az alkalmazott memóriát..... 1 pont
- 4) Összesen: 2 pont
 Két különböző számítógépet **hardver kompatibilisnek** nevezünk, ha az egyik számítógép *gépi kódját* a másik számítógép..... 1 pont
módosítás nélkül képes végrehajtani (értelmezni). 1 pont
- 5) e..... 1 pont
- 6) c..... 1 pont
- 7) Összesen: 3 pont
 A SMART technológia folyamatosan információkat gyűjt a meghajtókba épített érzékelőktől, 1 pont
 melyek alapján következtetni lehet a meghibásodásra. Ha a figyelt jellemző vagy jellemzők egy bizonyos értéket meghaladnak, akkor egy hang a betöltési folyamat közben jelzi, hogy célszerű lenne az adatokat egy másik adathordozóra átmenteni. 1 pont
 A figyelt jellemzők: bithibák számának növekedése, hőmérséklet emelkedése, az adatátviteli sebesség hirtelen csökkenése, stb (Legalább két jellemző megadása esetén adható a pont) 1 pont
- 8) I; H; I; I (minden helyes részválasz 1 pont) 4 pont
- 9) Összesen: 2 pont
 A jumper:.....
 a számítógép hardver elemein található két áramköri pontot összekötő eszköz. (apró elektronikus kapcsoló) 1 pont
 Használata:
 kézzel kell beállítani a működéshez szükséges állapotba 1 pont
- 10) Összesen: 2 pont
 A Plug and Play telepítést támogató *külső busz*. Az USB-t használva a számítógép leállítása és újraindítása nélkül lehet eszközöket a számítógéphez csatlakoztatni és arról leválasztani. 1 pont
 Egy USB-portra *több periféria csatlakoztatható* (127), amelyek között lehet PenDrive, hangszóró, telefon, CD-ROM meghajtó, botkormány, szalagos meghajtó, billentyűzet, képolvasó, kamera, fényképezőgép stb. 1 pont

Szoftver

- 11) Összesen: 3 pont
 Bekapcsoláskor működő öntesztelő program. Ez az első folyamat,
 amely a PC bekapcsolása után lefut. 1 pont
 Ellenőrzi, hogy az egyes részegységek (processzor, memória,
 grafikus kártya stb) működnek-e. 1 pont
 Általában a hiba kódjának megfelelő sípjelzésekkel jelzi a hiba
 okát, ha baj van. (Ha csak egyetlen rövid sípolás van, azt jelzi,
 hogy minden rendben van.) 1 pont
- 12) c. 1 pont
 13) c. 1 pont
 14) H; I; H; I (minden helyes részválasz 1 pont) 4 pont
 15) H; I; H; H (minden helyes részválasz 1 pont) 4 pont

Szövegszerkesztés, Táblázatkezelés:

- 16) b. 1 pont
 17) Összesen: 5 pont
B15: INDEX (A2 : A13 ; HOL . VAN (MIN (V2 : V13) ; V2 : V13 ; 0) ; 1)
 vagy
B15: INDEX (A2 : A13 ; HOL . VAN (MIN (V2 : V13) ; V2 : V13 ; 0))
 A helyes kitöltésért adható pontok:
 INDEX 2 pont
 HOL . VAN 2 pont
 MIN 1 pont
 Minden hibáért 1 pont levonás jár. (Negatív pontszám nem adható!)

Informatikai alapok:

- 18) H; I; H; I (minden helyes részválasz 1 pont) 4 pont
 19) d. 1 pont
 20) c. 1 pont

Hálózati ismeretek, HTML:

- 21) d. 1 pont
 22) A kereső motorok nem azonos keresési algoritmussal dolgoznak, 2 pont
 23) d. 1 pont
 24) A 2. sorban egy szabványosan megadott <a> tag (hivatkozás) található,
 amelynél a megjelenített szöveg és a hozzá tartozó hivatkozási cím ugyanaz ... 1 pont

Összesen: 50 pont

II. rész

- A feladatok megoldásához a számítógépes konfiguráción, illetve papíron és íróeszközön kívül egyéb segédeszközt nem használhat!
- Ahol a feladat szövege másképp nem rendelkezik, ott az adott feladatot megoldó program forráskódját kell beadnia! Amennyiben a megoldás egyéb fájlokat is használ (pl. adatbázisfájlok, indexfájlok, adatfájlok) természetesen azokat is be kell adnia.
- A feladatok megoldása során az aktuális szoftver jegyzékben előzetesen megadott programnyelvek közül az egyiket kell használnia. A 4. feladat megoldása során a választott adatbázis-kezelő rendszert, illetve az SQL nyelvet használhatja.
- A feladatok megoldása során, ahol ez külön nincs jelezve a feladatban, feltételezzük, hogy a program használója, a billentyűzetes inputoknál a megfelelő formátumú és a feladat kritériumrendszerének megfelelő értékeket ad meg, így *külön input ellenőrzéssel nem kell foglalkoznia*, ezért az ilyen jellegű kódrészekért többletpont sem adható.
- Ahol a feladat szövege a kimenet pontos formátumát nem határozza meg, ott törekedjen a kulturált, ám egyszerű megjelenítésre. A kiíratott adatok formátuma kellően olvasható legyen (pl. a valós számokat ne az exponenciális formátumban jelenítse meg stb.).
- A 4. feladat megoldását készítheti teljes egészében SQL nyelven is. Abban az esetben, ha a feladat az adatbázis létrehozását és feltöltését is előírja, az ezt végző SQL forráskódot is be kell adnia egy megfelelő szövegfájlban.
- A feladatok befejezésekor a vizsga helyszínén kiadott útmutatónak megfelelő helyre, a feladat sorszámának megfelelő elnevezésű alkönyvtárakba (FELADATn elnevezésű mappákba) mentse el az egyes feladatok megoldását adó forrásfájlt, illetve az esetlegesen szükséges egyéb kiegészítő állományokat.
(**FELADAT1...FELADAT4** könyvtárak!)

1. Feladat**10 pont**

A ferde hajítás távolságát az alábbi képlettel számoljuk ki:

$$s_{x \max} = \frac{v_0^2 \sin 2\alpha}{g},$$

ahol $s_{x \max}$ a hajítás távolsága méterben v_0 az elhajított test kezdősebességének nagysága m/s-ban, α a kezdősebesség irányvektorának a vízszintessel bezárt szöge, g pedig a gravitációs gyorsulás (melynek közelítő értéke $9,81 \text{ m/s}^2$ a Földön).

Készítsen programot, amely a képlet alapján meghatározza, hogy egy adott kezdősebességgel és adott szögben elhajított test milyen messzire repül! A kezdősebesség nagyságát és szögét a billentyűzetről olvassa be! Az eredményt a képernyőre írja ki!

Értékelés:

- a) A programkód szintaktikailag hibátlan 2 pont
 – Ezek a pontok csak abban az esetben adhatók meg, ha a programkód tartalmaz a b-e. szakaszokba tartozó, összességében legalább 3 pontot érő részmegoldást!
- b) Bemenő adatok tárolására alkalmas változók helyes deklarálása 2 pont
 – Kezdősebesség (valós típusú változó): 1 pont
 – Szög (valós típusú változó): 1 pont
- c) Az adatok beolvasása..... 2 pont
 – Kezdősebesség beolvasása: 1 pont
 – Szög beolvasása: 1 pont
 A pontok akkor adhatók meg, ha egyértelmű, hogy melyik adatot kell megadni!
- d) A távolság helyes kiszámítása 2 pont
 Nem osztható pontszám, akkor is megadható, ha az eredmény nem kerül változóba
- e) Az eredmény kiírása 2 pont
 – Az eredmény kiírásra kerül: 1 pont
 – A kiírás formátuma megfelelő (kerekítés): 1 pont

Mintamegoldás: (MS VISUAL STUDIO 2005 Express (VisualBASIC))

Module Module1

```

REM Ferde hajítás távolsága
Sub Main()
    REM Változók, konstansok deklarálása
    Const g As Double = 9.81
    Dim sxmax, v0, alfa, alfarad As Double

    REM Információk kiírása
    System.Console.WriteLine("A ferdén hajított test által megtett
távolság...")
    System.Console.WriteLine("-----")
    System.Console.WriteLine("A tizedes jel (./,) a Területi
beállításoktól függ!")
    System.Console.WriteLine("")

```

```

REM Bemeneti adatok beolvasása a billentyűzetről
System.Console.WriteLine("Kezdősebesség m/s (smax) = ")
v0 = System.Console.ReadLine()
System.Console.WriteLine("Hajítás szöge fokban (alfa) = ")
alfa = System.Console.ReadLine()
REM Konvertálás radiánra
alfarad = alfa * Math.PI / 180

REM Az eredmény kiszámítása
sxmax = v0 * v0 * Math.Sin(2 * alfarad) / g

REM Az eredmény megjelenítése
System.Console.WriteLine("A(z) " & v0 & " (m/s) kezdősebességgel "
& alfa & " fokos szögben elhajított tárgy " & Math.Round(sxmax, 2) & "
méterre repül.")
System.Console.WriteLine()

REM Programbefejezés
System.Console.WriteLine("Nyomj [enter]-t a befejezéshez!")
System.Console.ReadLine()
End Sub

End Module

```

2. Feladat

10 pont

Egy út mellé felállított traffipax készülék rögzíti a mellette elhaladó járművek sebességének nagyságát. Az adott helyen a megengedett legnagyobb sebesség 90 km/h. Készítsen programot, amely a billentyűzetről beolvassa a legutóbb elhaladt 10 jármű sebességét, majd képernyőre írja a szabályosan haladó járművek átlagsebességét, valamint a gyorsajtó járművek darab számát!

- a) A programkód szintaktikailag hibátlan..... 1 pont
 - Ez a pont csak abban az esetben adható meg, ha a programkód tartalmaz a b-d. szakaszokba tartozó, összességében legalább 3 pontot érő részmegoldást!
- b) Változók helyes deklarációja 2 pont
 - A sebességek tárolására alkalmas (valós elemű) tömb helyes deklarációja: 1 pont
 - Egyéb szükséges változók helyes deklarációja: 1 pont
- c) Az adatok beolvasása..... 2 pont
 - A 10 sebesség érték beolvasása megtörténik: 1 pont
 - A beolvasás folyamata jól nyomon követhető (kiírja, hogy hányadik adat beolvasásánál tart): 1 pont
- d) Eredmények meghatározása, kiírása..... 5 pont
 - A szabályosan haladók számának meghatározása: 1 pont
 - A szabályosan haladók sebességeinek az összegzése: 1 pont
 - A szabályosan haladók átlagsebességének a meghatározása: 1 pont
 - A gyorsajtók számának meghatározása: 1 pont
 - Minden eredmény kiírásra kerül, a kiírás egyértelmű és formátuma megfelelő: 1 pont

Mintamegoldás: (MS VISUAL STUDIO 2005 Express (VisualBASIC))
Module Module1

```
REM Sebessegek
Sub Main()
    REM Változók, konstansok deklarálása
    Const N As Integer = 10
    Const SebHat As Integer = 90
    Dim sebesseg(N), szabAtlag As Double
    Dim i, szabDb, gyorsDb As Integer

    REM Információk kiírása
    System.Console.WriteLine("Átlagsebesség és gyorsajtók...")
    System.Console.WriteLine("-----")
    System.Console.WriteLine("A tizedes jel (./,) a Területi
beállításoktól függ!")
    System.Console.WriteLine("")

    REM Bemeneti adatok beolvasása a billentyűzetről
    For i = 1 To N
        System.Console.Write("A(z) " & i & ". autó sebessége (km/h) =
")
        sebesseg(i) = System.Console.ReadLine()
    Next

    REM Az eredmény kiszámítása
    szabAtlag = 0
    szabDb = 0
    gyorsDb = 0
    For i = 1 To N
        If sebesseg(i) <= SebHat Then
            szabDb += 1
            szabAtlag += sebesseg(i)
        Else
            gyorsDb += 1
        End If
    Next
    szabAtlag /= szabDb

    REM Az eredmény megjelenítése
    System.Console.WriteLine("A szabályosan haladók átlagsebessége: " &
Math.Round(szabAtlag, 2))
    System.Console.WriteLine("A gyorsajtók száma: " & gyorsDb)
    System.Console.WriteLine()

    REM Programbefejezés
    System.Console.WriteLine("Nyomj [enter]-t a befejezéshez!")
    System.Console.ReadLine()
End Sub

End Module
```


3. Feladat**15 pont**

Egy pénzdíjas kerékpáros versenyen 36 induló van. A rajtszámok 1-től 36-ig kerülnek kiosztásra. A verseny 10 szakaszból áll. Minden szakasz végén rögzítjük, hogy melyik versenyző hányadikként zárta az adott szakaszt. Feltesszük, hogy kiesés és holtverseny semelyik szakaszon sem fordul elő. A verseny díjazására fordított összeg 1 millió forint. A díjat az a versenyző nyeri aki a legtöbb szakasz elsőséget szerezte. Amennyiben a legnagyobb számú szakasz elsőséget több versenyző is elérte a díjat közöttük egyenlő mértékben, forintra lefelé kerekítve osztják fel.

Készítsen programot az alábbi feladatok megoldására:

Töltse fel véletlenszerűen az egyes versenyzők egyes szakaszokon elért helyezését tároló változót! Ügyeljen arra, hogy egy adott szakaszon minden helyezési sorszám csak egyszer osztható ki!

Határozza meg a pénzdíjat nyert versenyzőt, illetve versenyzőket. A képernyőre írassa ki róluk az alábbi adatokat: (versenyző rajtszáma: megnyert pénzdíja)!

Az egyes versenyzőkre vonatkozó adatsort gömbölyű zárójelek közé zárja! Az adatsorokat külön sorokba írja!

Példa az eredmény kiírására.:

(2:333333)

(14:333333)

(23:333333)

- a) A programkód szintaktikailag hibátlan..... 1 pont
 – Ez a pont csak abban az esetben adható meg, ha a programkód tartalmaz a b-e. szakaszokba tartozó, összességében legalább 5 pontot érő részmegoldást!
- b) Változók helyes deklarációja 2 pont
 – A helyezések tárolására alkalmas adatszerkezet helyes deklarációja: 1 pont
 – Egyéb szükséges változók helyes deklarációja: 1 pont
- c) Az adatok véletlenszerű feltöltése 4 pont
 – A helyezési számok tárolására szolgáló változó teljesen feltöltésre kerül: 1 pont
 – Az egyes generált helyezési számok az előírt intervallumba esnek: 1 pont
 – Egy szakaszon belül nem fordul elő helyezési sorszám ismétlődés: 2 pont
- d) Eredmények meghatározása 6 pont
 – Legalább egy versenyző első helyezéseinek számát helyesen számítja: 1 pont
 – Minden versenyző első helyezéseinek számát helyesen számítja: 1 pont
 – Meghatároz egy olyan versenyzőt, aki a legtöbb első helyet érte el: 2 pont
 – Az összes nyertest jól meghatározza: 1 pont
 – A nyereségösszeget jól számítja ki: 1 pont
- e) Eredmények kiírása 2 pont
 – Minden nyertes sorszáma kiírásra kerül: 1 pont
 – A kiírás formátuma a példának megfelelő: 1 pont

Mintamegoldás: (MS VISUAL STUDIO 2005 Express (VisualBASIC))

```
Module Module1
    REM Bicikliverseny
    Structure eredmény
        Public rajtszam As Integer
        Public ido As Double
    End Structure
    Sub Main()
        REM Változók, konstansok deklarálása
        Const indulokSzama As Integer = 36
        Const szakaszokSzama As Integer = 10
        Dim dijazas As Integer = 1000000
        Dim helyezesek(indulokSzama, szakaszokSzama) As Integer
        Dim feltSeged(indulokSzama) As Integer
        Dim szakElsSzama(indulokSzama) As Integer
        Dim nyertesek(indulokSzama), nyertesekSzama, max As Integer
        Dim er(indulokSzama) As eredmény
        Dim i, j, k, l, s As Integer
        Dim autoRand As New Random()

        REM Információk kiírása
        System.Console.WriteLine("Bicikliverseny...")
        System.Console.WriteLine("-----")
        System.Console.WriteLine("")

        REM Bemeneti adatok generálása
        For i = 1 To szakaszokSzama
            For k = 1 To indulokSzama
                feltSeged(k) = k
            Next
            For j = 1 To indulokSzama
                s = autoRand.Next(indulokSzama - j + 1) + 1
                helyezesek(j, i) = feltSeged(s)
                For l = s To indulokSzama - j
                    feltSeged(l) = feltSeged(l + 1)
                Next
            Next
        Next

        REM Az eredmény kiszámítása / szakaszelsőségek megszámlálása
        For i = 1 To indulokSzama
            szakElsSzama(i) = 0
            For j = 1 To szakaszokSzama
                If helyezesek(i, j) = 1 Then
                    szakElsSzama(i) += 1
                End If
            Next
        Next

        REM Az eredmény kiszámítása / maximumérték és maximumhelyek meghatározása
        max = 0
        nyertesekSzama = 0
        For i = 1 To indulokSzama
            If szakElsSzama(i) > max Then
                nyertesekSzama = 1
                nyertesek(1) = i
            End If
        Next
    End Sub
End Module
```

```
        max = szakElsSzama(i)
    ElseIf szakElsSzama(i) = max Then
        nyertesekSzama += 1
        nyertesek(nyertesekSzama) = i
    End If
Next

REM Az eredmény megjelenítése
System.Console.WriteLine("A verseny nyertesei és díjaik...")
For i = 1 To nyertesekSzama
    System.Console.WriteLine("(" & nyertesek(i) & ":" &
Math.Truncate(dijazas / nyertesekSzama) & ")")
Next

REM Programbefejezés
System.Console.WriteLine("Nyomj [enter]-t a befejezéshez!")
System.Console.ReadLine()
End Sub

End Module
```

4. Feladat

15 pont

A feladatban szereplő adatbázis emberek közötti ismeretségeket nyilvántartó, kapcsolatépítő weblap adat tárolási rendszerének egyszerűsített modellje. Tároljuk benne a regisztrált felhasználók alapadatait, egymással való ismeretségi viszonyukat, valamint bizonyos, a rendszerben szereplő emberek által definiálható csoportokat illetve azokhoz való tartozásukat. Az adatbázisban a kapcsolatok felvétele oly módon zajlik, hogy egy már regisztrált felhasználó ismerősének jelöl egy másik szintén regisztrált felhasználót, majd az, a kapcsolatot visszaigazolja. Ezek után kerül a kapcsolat az adatbázis általunk ismert részébe. A regisztráció, jelölés, visszaigazolás folyamatának ábrázolása nem szerepel a feladat adatbázisában.

Az adatbázis elsősorban feladatkitűzési céllal készült, így természetesen nem modellezi tökéletesen a való életben felmerülő hasonló rendszerekkel kapcsolatos összes lehetséges helyzetet. A feladatokat az adott modell keretein belül kell megoldani.

A feladathoz mellékeljük a feltöltött minta adatbázist több formátumban, az alábbi fájlokban: ismeros.mdb, ember.txt, csoport.txt, csoportag.txt, ismer.txt .

Az adatbázis az alábbi táblákat (relációkat) tartalmazza:

(A „:” után az adott adat típusát adtuk meg, a „->” karakterek után pedig az esetlegesen meghatározott kapcsolatot. Az egyes tábláknál a kulcsot aláhúzott karakterekkel jelöljük.)

```

EMBER (
EAZON      : Egész szám          -> CSOPORT.LETREHOZO
                                     -> CSOPTAG.TAG
                                     -> ISMER.EMBER1
                                     -> ISMER.EMBER2

VNEV      : Szöveg
KNEV      : Szöveg
SZULDAT   : Dátum
NEM       : Szöveg
)

```

```

CSOPORT (
CSAZON     : Egész szám
CSNEV     : Szöveg
LETREHOZO : Egész szám   -> EMBER.EAZON
)

```

```

CSOPTAG (
CSTAZON   : Egész szám
CSOPORT   : Egész szám   -> CSOPORT.CSAZON
TAG       : Egész szám   -> EMBER.EAZON
)

```

```

ISMER (
IAZON     : Egész szám
EMBER1    : Egész szám   -> EMBER.EAZON
EMBER2    : Egész szám   -> EMBER.EAZON
)

```

Az **EMBER** tábla tartalmazza az egyes emberek azonosítóját (EAZON), vezetéknévét (VNEV), keresztnévét (KNEV), születési dátumát (SZULDAT) valamint a nemét (NEM).

A nem meghatározásánál az F jelöli a férfiakat és az N a nőket.

A **CSOPORT** tábla a felhasználók által definiált csoportok alapadatait tartalmazza. A csoport azonosítója (CSAZON), neve (CSNEV), és a létrehozójának EMBER táblai azonosítója (LETREHOZO) található a táblában.

A csoport tagságokat a **CSOPTAG** táblában tároljuk, ahol a kapcsolat azonosítója (CSTAZON), a csoport azonosítója (CSOPORT) és a csoport tagjának az EMBER táblában lévő azonosítója (TAG) található

Az emberek közötti ismeretségeket az **ISMER** tábla tartalmazza. Az ismeretség azonosítója (IAZON) mellett ebben a táblában még két mező található. A kapcsolatot megjelölő személy (EMBER1) és a kapcsolatot visszaigazoló személy (EMBER2) azonosítója. Ezen utóbbi két mező az EMBER táblában található EAZON mezőkkel kapcsolódik.

Értékelés, mintamegoldás:

Az itt megadott megoldásoktól eltérő helyes megoldások is elfogadhatóak. Ezek pontozásánál az ezen útmutatóban megadott megoldások pontozási arányait kell alkalmazni az esetleges rész megoldásoknál.

- a.) Készítsen lekérdezést, amely megadja a legidősebb úr vezeték és keresztnévét! Ügyeljen arra, hogy amennyiben több legidősebb hölgy is van (egy napon születettek) mindegyikük neve szerepeljen az eredményben!

```
SELECT vnev,knev
FROM ember
WHERE nem="F" and
szuldat=(SELECT MIN(szuldat) FROM ember WHERE nem="F");
```

1 pont
1 pont
3 pont

- b.) Készítsen lekérdezést a csoport tagságokról! A lekérdezés eredménye tartalmazza az összes csoport tagságot. Az eredmény egy sora adja meg a csoportnevet és mellette az adott tag vezeték és keresztnévét. A lista legyen rendezet csoportnév szerint, csoporton belül vezetékneve azon belül pedig keresztnév szerint!

```
SELECT csnev, vnev,knev
FROM csoport,csoptag,ember
WHERE (csazon=csoport) and (tag=eazon)
ORDER BY 1,2,3;
```

1 pont
1 pont
2 pont
1 pont

- c.) Készítsen listát az összes regisztrált személy összes ismeretségéről! (A lista tehát minden ismeretséget kétszer jelenít meg egyszer az egyik ismerősnél egyszer, pedig a másiknál.) A lista tartalmazza a NEV1 oszlopban az egyik ismerős nevét a NEV2-ben pedig a másikat! A lista legyen NEV1-szerint rendezett, a NEV1 szerint azonosaknál pedig NEV2 szerint!

```
SELECT e1.vnev+" "+e1.knev as nev1, e2.vnev+" "+e2.knev as nev2
FROM ismer,ember as e1, ember as e2
WHERE ( e1.eazon=ismer.ember1 and e2.eazon= ismer.ember2) or
( e1.eazon=ismer.ember2 and e2.eazon= ismer.ember1)
ORDER by 1,2;
```

1 pont
1 pont
1 pont
1 pont
1 pont

Összesen: 50 pont