

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

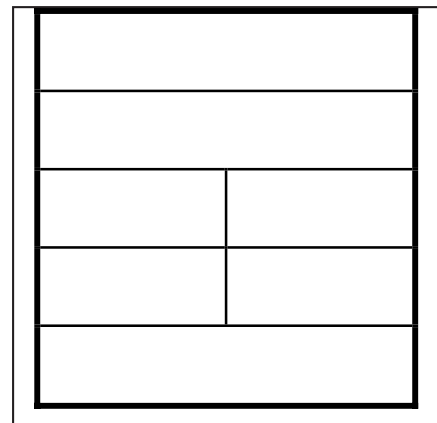
## 1. Egybesült karaj

Egyes divatos éttermek a weblapjaikon recepteket is bemutatnak, amelyeket akár otthon is el lehet készíteni. Készítsen weblapot, és hozzá egy fejlécképet a minta és a leírás alapján, amely bemutatja az egyik kedvelt húsétel elkészítését!

A weboldal és az azon szereplő fejléckép elkészítéséhez szükséges képállományok a *karaj.png*, *eszkozok.png* és *szakacs.png*. A weblap szövegét a *recept.txt* (UTF-8 kódolású) állomány tartalmazza.

A weblapot stílusok nélkül vagy stílusokkal is elkészítheti. Ha stílusokkal készíti, akkor használhatja a *stilus.css* stíluslapot, amelyben még további beállításokat kell tennie.

- Készítse el a weblapra beillesztendő képet az *eszkozok.png* és a *szakacs.png* állományok felhasználásával. Az elkészített képállomány neve *fejlec.png* legyen!
  - Az elkészített kép 800×120 képpont méretű legyen!
  - A kép háttéréhez sötétbarna RGB(80, 59, 53) kódú színt alkalmazzon!
  - Szűrje be a *szakacs.png* képet egy, és az *eszkozok.png* képet három példányban átlátszó háttérrel a mintához hasonló elrendezésben! A rajzok egymáshoz és a kép széléhez ne érjenek hozzá!
  - Az evőeszközök rajzára, a szöveg háttérének helyezzen el egy, a kép háttérszínének megfelelő sötétbarnán kitöltött téglalapot úgy, hogy a villák fogai és a nyelek alján levő lyukak látszódjanak még!
  - A felirat elkészítéséhez világosbarna RGB(244, 198, 154) kódú színt alkalmazzon! Írja a minta szerinti helyre az „Arany Fakanál Bisztró” feliratot! A felirat a téglalap alsó és felső széléhez ne érjen hozzá, de legalább a téglalap magasságának a fele legyen a betűméret!
- Az oldalarányok megtartásával a *karaj.png* kép méretét módosítsa 300 képpont szélességűre, és mentse *karajkep.png* néven!
- Készítse el a *sult.html* állományt! Az oldal forrásszövegét a *recept.txt* állományban találja. A böngésző címsorában megjelenő cím szövege: „Recept” legyen!
- Az oldal háttérszíne legyen fekete, és a tartalom formázásához három színt alkalmazzon: sötétbarnát (#503B35 kódú szín), világosbarnát (#F4C69A kódú szín) és sötétnarancsot (#F26600 kódú szín)!
- A weblap tartalma egy sötétbarna téglalapon jelenjen meg! A weblap elrendezését az ábrának megfelelően alakítsa ki táblázattal, vagy a stíluslapban lévő osztályok beállításával! A sötétbarna téglalap legyen 800 képpont széles, az oldalon vízszintesen középre igazított és szegély nélküli! Minden szöveges tartalom esetén a belső margó 10 képpont legyen!
- Az oldalon a linkek színe minden állapotban sötétnarancs, és a szöveg színe világosbarna legyen!
- Szűrje be a *fejlec.png* képet a minta szerinti helyre! A képnek ne legyen szegélye! Ha a képet nem sikerült elkészítenie, akkor használja a *potfejlec.png* állományt!





--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--


8. Szúrjon be a fejléckép után egy 60% szélességű vízszintes vonalat középre!
9. A weblap teljes szövege Verdana betűtípusú legyen!
10. A `recept.txt` állományból illessze be a szöveget a weblapba! Az oldal címét, az „**Egybesült karaj recept**” szöveget alakítsa egyes szintű címsor, és a másik öt alcímet kettes szintű címsor stílusúvá! Állítsa be, hogy minden cím és alcím sötétnarancs legyen!
11. Hozza létre a mintán látható felsorolásokat a megfelelő típussal!
12. Szúrja be a „**Hozzávalók / 6 adag**” cím mellé a `karajkep.png` képet az oszlopában vízszintesen középre! Állítsa be, hogy ha az egeret a képre mozgatjuk a böngészőben, akkor magyarázó szöveg jelenjen meg a buboréksúgóban! Ehhez a kép megjelenítéséért felelős taget egészítse ki a következő paraméterrel: `title="A sült húst szeleteljük!"`
13. Állítsa be a szöveg tördelését és formázását a mintának megfelelően!
14. A „**tűzdelt karaj**” és a „**sertéssült**” szavakat alakítsa linkké, amelyek a kapcsos zárójelekben lévő címekre mutassanak! A kapcsos zárójeleket a címekkel együtt törölje ki!

30 pont

Minta:



## Egybesült karaj recept



### Hozzávalók / 6 adag

- 1,5 kg sertés rövidkaraj
- 2 gerezd fokhagyma
- 2 ek sertézsír (olvasztva)
- 1 ek köménymag
- 1 ek fűszerpaprika
- 20 dkg bacon
- 3 db közepes vöröshagyma
- só ízlés szerint
- bors ízlés szerint

#### Sztori

Egy jól elkészített egészben sült hús remek kísérője lehet egy főzeléknek, vagy bármilyen más köretnek, másnap pedig kitűnő adalék egy jó szendvicshöz. Sok probléma pedig nincs az elkészítésével, csak idő kell hozzá. Próbáljátok ki Ti is!

#### Alapanyagok

bacon, fokhagyma, fűszerpaprika, köménymag, sertés rövidkaraj, sertézsír, só, vöröshagyma

#### Egyéb elnevezések

[tűzdelt karaj](#), [sertéssült](#)

#### Elkészítés

1. Fogjuk a karajt, és egy éles kés hegyével jó pár helyen megszúrkaljuk. A kapott kis "nyílásokba" töltjük a nagyobb szeletekre vágott fokhagymát. Majd jól besózzuk a húst.
2. A megolvasztott zsírt összekeverjük a köménymaggal és fűszerpaprikával, és alaposan bekenjük vele a húst, és hagyjuk 1-2 órát kicsit pácolódni.
3. Első körben 220 fokra előmelegített sütőbe tesszük a húst fedés nélkül kb. 20 percre. Ezután kivesszük, és hagymanegyedekkel körberakjuk a húst, majd lefedjük, és most már csak 180 fokra sütőbe tesszük vissza 1 órára.
4. Ha még szükséges egy kis piritás a húsról, akkor a legvégén 10 percet fedés nélkül rápirítunk.

- sütési hőfok: 180 °C
- sütési mód: alul-felül sütés
- költség: 2000 Ft

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

## 2. Robotverseny

A robotok építése és programozása egyre népszerűbb a diákok körében, így egyre több hazai és nemzetközi robotversenyt rendeznek. Egy ilyen verseny eredménylistáját kell feldolgoznia ebben a feladatban. Az adatok az UTF-8 kódolású *eredmenyek.txt* állományban találhatók.

Feladata az adatok elemzése és megjelenítése táblázatkezelő program segítségével!

*A megoldás során vegye figyelembe a következőket!*

- *Segédszámításokat a Q oszloptól jobbra végezhet. Amennyiben lehetséges, a megoldás során képletet, függvényt, hivatkozást használjon, hogy az alapadatok módosítása esetén is a kívánt eredményeket kapja!*
- *A részfeladatok között van olyan, amely egy korábbi kérdés eredményét használja fel. Ha a korábbi részfeladatot nem sikerült teljesen megoldania, használja a megoldását úgy, ahogy van, vagy írjon be egy valószínűnek tűnő eredményt, és azzal dolgozzon tovább! Így ugyanis pontokat kaphat erre a részfeladatra is.*

1. Töltse be a tabulátorokkal tagolt, UTF-8 kódolású *eredmenyek.txt* szövegfájl a táblázatkezelő program munkalapjára az A1-es cellától kezdődően! Munkáját a táblázatkezelő alapértelmezett formátumában *robotverseny* néven mentse!

A versenyen a robotok négy alkalommal indulhatnak el a pályán. Az eredmény az egyes fordulóknál elért pontszámoktól és a feladat megoldásához szükséges időtől függ. A pontszámból a nagyobb, az időeredményből – azonos pontszám esetén – a kisebb eredmény a jobb.

Az A1:H26 tartományban az egyes fordulóknál kapott pontszámokat találja. Az A28:G53 tartományban a feladatok végrehajtására fordított időt látja fordulónként, másodpercben megadva. Ha az időeredményből a 0 érték van, az azt jelenti, hogy valamilyen ok miatt a robot el sem tudott indulni. Ha egy robot egyik fordulóban sem tudott elindulni, akkor nem kerülhetett be a táblázatba.

2. Az F2:F26 tartomány celláiban határozza meg az egyes csapatok által elért legnagyobb pontszámot!
3. A G2:G26 tartomány celláiban adja meg az egyes csapatok négy forduló alatt elért összesített pontszámát! A G29:G53 tartomány celláiban pedig csapatonként határozza meg a négy forduló összesített időértékét!
4. A B29:E53 tartományban feltételes formázással emelje ki szürke háttérrel a 0 értéket tartalmazó cellákat!
5. Az F29:F53 tartomány celláiban adja meg – másolható képlet segítségével – csapatonként a négy forduló közül a legkisebb időt! A legkisebb időérték meghatározásánál a 0 másodperces időértékeket figyelmen kívül kell hagyni.
6. A rangsor megállapításához a négy forduló közül a három legnagyobb pontszámút veszik figyelembe. A „**versenypont**” felirat alatt a H2:H26 tartomány celláiban csapatonként adja meg a három legjobb eredményű forduló összesített pontszámát!
7. A végeredmény meghatározásához a versenypontokat és az összesített időértékeket foglalja össze a J1:M26 segédtáblázatban! Képletek segítségével határozza meg a K oszlopban felsorolt csapatok nevéhez tartozó megfelelő értékeket! Az L2:L26 tartomány celláiba a versenypont értékei, az M2:M26 tartomány celláiba pedig az összesített időértékek kerüljenek! Másolható képleteket használjon!

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

8. A helyezések elsődlegesen versenypontok alapján alakulnak ki. Azonos pontszámú csapatok között az összesített idő dönt. Ezeknek a szempontoknak a figyelembevételével rendezze a *J2:M26* tartomány sorait! Legfelül legyen az első helyezett!
9. Szeretnénk tudni, hogy hány csapat ért el egy megadott pontszámot. Az *O2* cellába írjon egy tetszőleges pontértéket, majd a *P2* cellába írjon olyan képletet, ami megadja, hogy hány csapat érte el vagy haladta meg a megadott versenypont értéket!
10. Az *A1:H1*, *J1:M1*, *O1:P1* és *A28:G28* tartományban lévő cellák háttere legyen szürke! A *B1:E1* tartomány celláit és a *B28:E28* tartomány celláit vonja össze! A *B29:G53* tartomány celláiban az időértékek számformátuma a mintának megfelelően jelenjen meg! A munkalapon úgy állítsa be az oszlopszélességeket, hogy minden adat olvasható legyen!

15 pont

**Minta:**

	A	B	C	D	E	F	G	H
1	csapat neve	pont			legnagyobb		összesen	versenypont
2	LeGo	140	110	110	140	140	500	
3	DreamTeam	0	5	10	0	10	15	
4	Kockák	0	0	60	45	60	105	
5	Krumpli	0	0	20	0	20	20	
6	Robotolók	0	60	45	85	85	190	
7	IRT	0	20	85	40	85	145	

	A	B	C	D	E	F	G
28	csapat neve	idő			legkisebb		összesen
29	LeGo	101 s	102 s	102 s	102 s	101 s	101 s
30	DreamTeam	5 s	10 s	11 s	5 s	5 s	5 s
31	Kockák	9 s	7 s	61 s	64 s	7 s	9 s
32	Krumpli	16 s	40 s	24 s	52 s	16 s	16 s
33	Robotolók	12 s	62 s	73 s	66 s	12 s	12 s
34	IRT	10 s	51 s	65 s	4 s	10 s	10 s

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

### 3. Közösségi szolgálat

Néhány éve az érettségi vizsga megkezdésének feltétele, hogy a diákok összesen 50 óra közösségi szolgálatot teljesítsenek. A közösségi szolgálat keretében többféle munkát végezhetnek a diákok. A legtöbb iskolában igyekeznek elérni, hogy a tanulók az 50 órát már az érettségi éve előtt teljesítsék. Ebben a feladatban egy iskola 2016/2017-os tanévre vonatkozó adatait dolgozzuk fel.

1. Készítsen új adatbázist *kozossegi* néven! A mellékelt négy – tabulátorokkal tagolt, UTF-8 kódolású – szöveges állományt (*diak.txt*, *jelentkezes.txt*, *munka.txt*, *tevekenyseg.txt*) importálja az adatbázisba a fájl névvel azonos néven (***diak***, ***jelentkezes***, ***munka***, ***tevekenyseg***)! Az állományok első sora a mezőneveket tartalmazza. A létrehozás során állítsa be a megfelelő típusokat és az elsődleges kulcsokat!

#### Táblák:

***diak*** (*id*, *nev*, *osztaly*)

<i>id</i>	A diák azonosítója (szám), ez a kulcs
<i>nev</i>	A diák neve (szöveg). Az iskolában minden diák neve egyedi, ezt a feladat megoldása során kihasználhatja.
<i>osztaly</i>	A diák osztálya (szöveg)

***jelentkezes*** (*diakid*, *munkaid*, *ervenyes*, *elfogadva*, *teljesitve*)

<i>diakid</i>	A diák azonosítója (szám), az összetett kulcs része
<i>munkaid</i>	Annak a munkának az azonosítója, amelyre a diák jelentkezett (szám), az összetett kulcs része
<i>ervenyes</i>	A diák jelentkezésének érvényessége (logikai); értéke hamis, ha a diák lemondta a jelentkezését
<i>elfogadva</i>	Azt jelzi, hogy a jelentkezést elfogadták-e (logikai); értéke igaz, ha elfogadták, hamis visszautasítás esetén
<i>teljesitve</i>	A mező a munka teljesítését rögzíti (logikai); értéke a teljesített munka esetén igaz

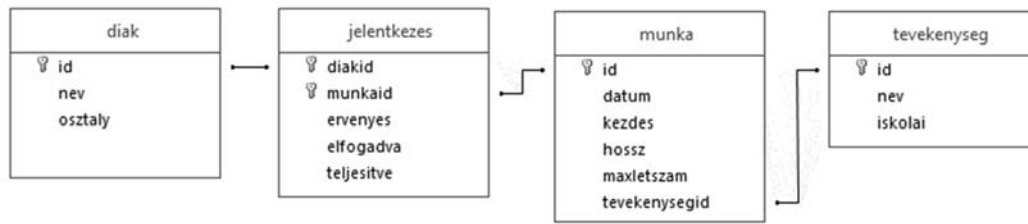
***munka*** (*id*, *datum*, *kezdes*, *hossz*, *maxletszam*, *tevekenysegid*)

<i>id</i>	A munka azonosítója (szám), ez a kulcs
<i>datum</i>	A munkavégzés dátuma (dátum)
<i>kezdes</i>	A munkakezdés ideje az adott napon (idő)
<i>hossz</i>	A munkavégzés órában mért hossza (szám)
<i>maxletszam</i>	A munkára igényelt maximális létszám (szám)
<i>tevekenysegid</i>	A tevékenység azonosítója (szám)

***tevekenyseg*** (*id*, *nev*, *iskolai*)

<i>id</i>	A tevékenységi terület azonosítója (szám), ez a kulcs
<i>nev</i>	A tevékenységi terület neve (szöveg)
<i>iskolai</i>	A tevékenységi terület az iskolához kötött-e (logikai); értéke igaz, ha a diák saját iskolájában végzi

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--



A következő feladatok megoldásánál a lekérdezéseket és a jelentést a zárójelben olvasható néven mentse! Ügyeljen arra, hogy a megoldásban pontosan a kívánt mezők szerepeljenek!

- Készítsen lekérdezést, amely ábécérendben megjeleníti az iskolához kötött tevékenységi területek nevét! (**2iskolai**)
- Készítsen lekérdezést, amely megadja, hogy a tanév során összesen hány órányi feladat volt elérhető a diákok számára! A lekérdezés készítése során vegye figyelembe a létszámot is! (**3oraszam**)
- Készítsen jelentést, amely kilistázza a 10. évfolyam tanulói által ténylegesen teljesített munkák adatait! A munkákat osztály, azon belül diákok szerint csoportosítva, időrendben jelenítse meg! A szövegszerű tartalmat tekintve az alábbi minta legyen a meghatározó! Biztosítsa, hogy minden érték látható legyen! A jelentést lekérdezéssel készítse elő! (**4evf10**)

A 10. évfolyamos diákok közösségi munkája a 2016/2017-es tanévben

osztály	név	dátum	időpont	óraszám	tevékenység
10/A	Ábrahám Katalin	2016. 11. 19.	15:00:00	3	könyvtárrendezés
		2016. 12. 09.	16:30:00	2	óvodai munka
		2016. 12. 18.	8:30:00	2	óvodai munka
		2017. 04. 23.	8:30:00	2	óvodai munka
		2017. 05. 20.	8:30:00	3	tolmácsolás
		2017. 06. 03.	9:00:00	3	állatkerti felügyelet
		2017. 06. 19.	10:30:00	3	táboroztatás
		2017. 06. 28.	8:00:00	4	gyermekfelügyelet
		2017. 08. 13.	8:30:00	2	óvodai munka

- Készítsen lekérdezést, amely kilistázza azon diákok nevét, akikkel legalább kétszer előfordult, hogy jelentkezésüket elfogadták, de nem teljesítették a választott feladatot! A diák nevét és a távolmaradások számát jelenítse meg! (**5tobbszor**)
- Készítsen lekérdezést, amely megadja azokat az őszi szünetre eső munkákat, amelyekre még nem volt jelentkező, azaz a **jelentkezés** táblában nem szerepelnek! A napot, a kezdési időt, a hosszt és a tevékenység nevét dátum, azon belül kezdési idő szerint rendezve jelenítse meg! 2016-ban az őszi szünet első napja október 29., utolsó napja november 6. volt. (**6senki**)
- Készítsen lekérdezést, amely osztályonként megadja azon diákok számát, akik teljesítettek már közösségi munkát! (**7stat**)

30 pont

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

## 4. Sorozatok

Sok olyan sorozatrajongó van, aki folyamatosan követi a kedvelt sorozatait. Egy, az angol nyelvű sorozatokért rajongó személy feljegyzést készített egy nyolc hónapos időszak kedvenc sorozatairól.

A `lista.txt` fájl a rajongó által kedvelt sorozatok adásba kerülésének dátumát, a sorozat angol címét, az évadot és az epizód számát, az epizód hosszát percben és egy jelzést tartalmaz, hogy a lista készítője megnézte-e már azt az epizódot. Ezek az adatok egymás után külön sorokban szerepelnek. A fájlban biztosan 400-nál kevesebb epizódról van adat, epizódonként 5 sorban.

A példában látható, hogy a `Puzzles` című sorozat 3. évadának 10. epizódja 2018. 01. 19-én került adásba. Az epizód 43 perces, és még nem nézte meg a lista készítője.

- A dátumokat mindig „éééé.hh.nn” formátumban rögzítették. Vannak olyan sorozatrészek, amelyeknek a lista rögzítésekor még nem tudták az adásba kerülésük idejét. Ezeknél a dátum helyett mindig az „NI” rövidítés szerepel.
- Az évad jelzése vezető nullák nélkül történik, az epizód számát pedig mindig két számjeggyel rögzítették. Az évad és az epizód számát egy „x” választja el egymástól.
- Az egyes sorozatok epizódjai mindig ugyanolyan hosszúak.
- Az epizóddal kapcsolatos utolsó adat értéke „0” vagy „1”. Az 1-es számjegy jelöli, hogy az adott részt már megnézte a lista készítője, a 0 pedig azt, hogy még nem látta.

Például:

```
...
2018.01.19
Puzzles
3x10
43
0
NI
Puzzles
3x11
43
0
...
```

Készítsen programot a `lista.txt` állomány adatainak feldolgozására! A program forráskódját mentse `sorozatok` néven! (A program megírásakor a felhasználó által megadott adatok helyességét, érvényességét nem kell ellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.)

A képernyőre írást igénylő részfeladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például `2. feladat:`)! Ha a felhasználótól kér be adatot, jelenítse meg a képernyőn, hogy milyen értéket vár! Az ékezetmentes kiírás is elfogadott.

1. Olvassa be és tárolja el a `lista.txt` fájl tartalmát!
2. Írassa ki a képernyőre, hogy hány olyan epizód adatait tartalmazza a fájl, amelynek ismert az adásba kerülési dátuma!
3. Határozza meg, hogy a fájlban lévő epizódok hány százalékát látta már a listát rögzítő személy! A százaléértéket a minta szerint, két tizedesjeggyel jelenítse meg a képernyőn!
4. Számítsa ki, hogy összesen mennyi időt töltött a személy az epizódok megnézésével! Az eredményt a minta szerint nap, óra, perc formában adja meg!
5. Kérjen be a felhasználótól egy dátumot „éééé.hh.nn” formában! Határozza meg, hogy az adott dátumig megjelent epizódokból melyeket nem látta még! Az aznapi epizódokat is számolja bele! A feltételnek megfelelő epizódok esetén írja a képernyőre tabulátorral elválasztva az évad- és az epizódszámot, valamint a sorozat címét a minta szerint!



--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

6. Készítse el az alábbi algoritmus alapján a hét napját meghatározó függvényt! A függvény neve *Hetnapja* legyen! A függvény az év, hónap és nap megadása után szöveges eredményként visszaadja, hogy az adott nap a hét melyik napja volt. (Az  $a$  és  $b$  egész számok maradékos osztása esetén az  $a \div b$  kifejezés adja meg a hányadost, az  $a \bmod b$  pedig a maradékot, például  $17 \div 7 = 2$  és  $17 \bmod 7 = 3$ .)

```
Függvény hetnapja(ev, ho, nap : Egész) : Szöveg
    napok: Tömb(0..6: Szöveg)= ("v", "h", "k", "sze",
                                "cs", "p", "szo")
    hónapok: Tömb(0..11: Egész)= (0, 3, 2, 5, 0, 3, 5, 1, 4, 6, 2, 4)
    Ha ho < 3 akkor ev := ev - 1
    hetnapja := napok[(ev + ev div 4 - ev div 100 +
                      ev div 400 + hónapok[ho-1] + nap) mod 7]
Függvény vége
```

7. Kérjen be a felhasználótól egy napot az előző feladatban látható rövidített alakban! A napokat egy (h, k, p, v), kettő (cs), vagy három (sze, szo) karakterrel adja meg! Határozza meg, hogy a fájlban lévő sorozatok közül melyike(ke)t vetítik az adott napon! A sorozatok nevét a minta szerint jelenítse meg a képernyőn! Ha az adott napon egy sorozatot sem adtak adásba, akkor „Az adott napon nem kerül adásba sorozat.” üzenetet jelenítse meg!
8. Határozza meg sorozatonként az epizódok összesített vetítési idejét és az epizódok számát! A számításnál vegye figyelembe a vetítési dátummal nem rendelkező epizódokat is! A megoldás során felhasználhatja, hogy egy sorozat epizódjainak adatai egymást követik a forrásállományban. A listát írja ki a *summa.txt* fájlba! A fájl egy sorában a sorozat címe, az adott sorozatra vonatkozó összesített vetítési idő percben és az epizódok száma szerepeljen szóközzel elválasztva!

45 pont

#### Minta a szöveges kimenetek kialakításához:

```
2. feladat
A listában 202 db vetítési dátummal rendelkező epizód van.

3. feladat
A listában lévő epizódok 45,66%-át látta.

4. feladat
Sorozatnézéssel 2 napot 15 órát és 32 percet töltött.

5. feladat
Adjon meg egy dátumot! Dátum= 2017.10.18
7x01    The Fable
7x02    The Fable
15x04   Military Police
5x03    Spy School
5x04    Spy School
4x04    The Elite Minds

7. feladat
Adja meg a hét egy napját (például cs)! Nap= cs
The Hospital
Spectacular Power
Upper Story
Chicago Flame
Shrinktime
```

#### Minta a *summa.txt* fájl kialakításához:

```
Games 420 7
The Fable 588 14
The IT Guy 450 10
```