„TÁMOP-4.1.2/A/1-11/1-2011-0015 Egészségügyi Ügyvitelszervező Szakirány: Tartalomfejlesztés és Elektronikus Tananyagfejlesztés a BSc képzés keretében”

Irodai programcsomagok használata és programozása

e-Book

Németh Felicián

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL

A projektek az Európai Unió támogatásával valósulnak meg.

Tartalom

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

Ábrák jegyzéke

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

1. Bevezető

Feltételezett előismeretek. Irodai programcsomagok a mindennapi munkában. Mikor érdemes a manuális munkát kiváltani automatizmussal?

Jelen tankönyv célja kettős. Egyrészt haladó szintű felhasználói ismeretek kíván átadni az irodai programcsomagok (szövegszerkesztő, táblázatkezelő, prezentációszerkesztő) használatában, különös tekintettel a táblázatkezelőre. Másrészt bevezetést kíván nyújtani e programok automatizálásába az általuk felkínált (makró-) nyelven való programozásukkal. A tárgyalt tananyag ennek megfelelően készség szintű gyakorlati ismereteket nyújt egyszerű Office programozási feladatok kivitelezésére Visual Basic nyelven. Az olvasóról feltételezi, hogy kisebb feladatok elvégzésére már használta az Excel, Word programokat, és hogy programozási alapismeretekkel is rendelkezik.

A tankönyvek azonban nem célja részletes, mindenre kiterjedő ismeretek átadni a tárgyalt programokról vagy akár magáról a Basic nyelvről sem. Sőt, a Basic programnyelvről és az Office alkalmazások programozási felületéről a lehető legkevesebb szó esik. A cél az, hogy egy olyan, kevés elemből álló eszközkészletet sajátítsunk el, amivel sok gyakorlati feladat megoldható. Valamint, hogy megtanuljuk, hogyan tudjuk a kis fegyvertárunkat külső segítség nélkül magunk is bővíteni.

Ha gyakran kell dolgozunk egy eszközzel, akkor érdemes energiát szánni arra, hogy elsajátítsuk az eszköz használatának minden csinyját-binját. A befektetett energia hosszú távon megtérül. Az irodai programok esetén a gyakran ismétlődő manuális munkát kiválthatjuk egy jól megírt makróval, ami a feladatot gyorsan és hibamentesen el tudja végezni. Érdemes-e azonban belefogni egy makró leprogramozásának, amikor a makró megírása több időt vesz igénybe, mint a feladat kézi elvégzése? Paradoxonnak tűnhet, de ha nem sokkal lassabb az automatikus feladatmegoldás leprogramozása, még akkor is érdemes azt választanunk. Egyrészt az automatikus megoldás megírásából tanulunk, a következő hasonló programozási feladatnak már nagyobb rutinnal fogunk nekikezdeni, így jó eséllyel azt hamarabb fogjuk befejezni. Tehát ha mostani feladatot gyorsabb manuálisan elvégeznünk, hosszú távon itt is nyerünk az automatikus megoldás előkészítésével. Továbbá míg a kézi feladatmegoldás szinte mindig monoton, unalmas munkát jelent, amiben frusztrációt okoz, mert könnyű hibázni benne. Ezzel szemben egy, nem megoldhatatlanul nehéz programozási feladat kellemes szellemi kihívást jelent és a sikeres megoldás remélhetőleg jól eső sikerélményt okoz.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

A tankönyv általánosabb célja tehát, hogy az olvasó az anyag elsajátítása során rutinra tegyen szert az algoritmikus gondolkodást igénylő problémamegoldásban.

2. Szakdolgozat készítése Wordben

Stílusok. Ábrák, táblázatok (beillesztés, képaláírások, ábrák hivatkozása a szövegben). Tartalom- és irodalomjegyzék készítése

Noha a Word dokumentumszerkesztővel gyorsan, különösebb előismeretek nélkül elkészíthetünk jól kinéző pár oldalas műveket, egy nagyobb lélegzetvételű dokumentum megírása során érdemes betartani néhány szerkesztési elvet és tudatosan használni a Word egyes funkcióit.

Az egyik ilyen elv a stílusok használata. Ahelyett hogy a szöveget közvetlenül formáznánk meg, stílusokon keresztül állítjuk be a dokumentum kinézetét. Tehát a dokumentum készítése során egy szövegrésznek csak a logikai struktúrában elfoglalt helyét adjuk meg, például azt, hogy címsor-e vagy a normál folyószöveghez tartozik-e, és nem azt, hogy milyen betűtípussal és betűmérettel jelenjen meg. Ennek az elvnek az egyik előnye, hogy a dokumentum írása során könnyebben tudunk annak tartalmára koncentrálni, és a formázás nem vonja el figyelmünket a tartalom előállításától. Másrészről könnyebb egységesen kinéző dokumentumok írni, mivel a külalakot csak a stílusok szabják meg. Harmadrészt pedig sokkal könnyebb a dokumentum kinézetét utólag módosítani: tegyük fel, hogy a mondaton belüli kiemelésre félkövér betűtípust használtuk és ezt szeretnénk lecserélni dőltre. Az összes kiemelés cseréje rendkívül időigényes lehet, ha a szöveg betűtípusát közvetlenül változtattuk meg és a félkövér betűtípussal a szövegközi kiemelésen kívül mást is jelöltünk. Ezzel szemben egyszerűen csak a kiemelés nevű stílus betűtípusát kell egyelten egyszer módosítani, ha előzetesen ezt a stílust használtuk a szövegközi kiemelésekre.

Ha nem vagyunk hozzászokva, hogy a közvetlen formázás helyett csak a stílusokat használjuk, akkor érdemes a Korrektúra szalag Védelem ikonján keresztül bekapcsolni a Formázás és módosítás korlátozása funkciót.

Szakdolgozatnál és más hosszabb szakmai szövegnél elvárt, hogy a műben szereplő ábrák (és táblázatok) sorszámozottak legyenek, a folyószövegben ezzel a sorszámmal kell hivatkozni az ábrákra. Az ábrát és a hozzátartozó képaláírást úgy érdemes elkészíteni, hogy egy lelkes olvasó pusztán az ábra tanulmányozása és a képaláírás elolvasása után megértse az ábrát, illetve hogy értelmes

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

következtetéseket tudjon levonni. Ennek ellenére a folyószövegben is szükséges elemezni az ábrát, fel kell hívni az olvasó figyelmét az ábrán látható fontosabb részletekre.

Wordben képet a Beszúrás szalag Ábrák csoportjának Kép ikonján keresztül tudunk beszúrni. Ezekután a beszúrt kép felugró menüjét jobbklikkel tudjuk elérni, ahol a képaláírás megadására van lehetőségünk. Ahogy az a 2-1. ábrán is látszik, ezen a ponton némi nehézségbe ütközhetünk, ha az elsődleges dokumentumnyelv nem magyar. Ha van rá lehetőségünk, akkor a Start menü, Programok, Microsoft Office, Microsoft Office-eszközök, Microsoft Office Nyelvi beállítások útvonalon keresztül tudjuk elindítani azt a programot, amivel az elsődleges dokumentumnyelvet be lehet állítani. Mindenesetre azt szeretnénk, hogy így vagy úgy, de a "Figure 1" helyett "1. ábra" szerepeljen a képaláírás elején. Ennek legegyszerűbb módja, ha a feliratot kihagyjuk a képaláírásból és az ábra sorszáma után kézzel írjuk oda azt, hogy "ábra". Itt ami sokakat megzavar az az, hogy a "felirat" szó nem a teljes képaláírást jelenti, tehát nem azt, például azt hogy "2-1. ábra. Képaláírás beszúrása angol nyelvű környezetben.". A "felirat" ennél az ablaknál csak a sorszámozott elem kategóriájának nevét jelenti, azaz jelen esetben a "Figure"-t.

2-1. ábra. Képaláírás beszúrása angol nyelvű környezetben.

A Hivatkozás szalag Feliratok csoportjának Kereszthivatkozások ikonjával tudunk beszúrni a folyószövegből egy hivatkozást egy ábrára. A felugró ablakban a hivatkozás típusánál tudjuk kiválasztani, azaz azt, hogy melyik feliratcsoportból szeretnénk kiválasztani azt az elemet, amihez a hivatkozást be kívánjuk szúrni. Azért is volt szerencsés kihagyni a feliratot a képaláírásból, mert most amikor a

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

szövegből hivatkozunk egy ábrára a „Csak címke és szám”1 típusú beszúrásakor csak a sorszámot szúrja be nekünk a Word, így többféleképpen tudjuk a hivatkozást mondatba helyezni, tehát az „1. ábra azt mutatja...” helyett írhatjuk például azt, hogy az „1. ábráról leolvasható…”.

Mi értelme automatikusan generált sorszámokat használni és a folyószövegben az ábrákra ezekkel a sorszámokkal hivatkozni? Hosszabb dokumentum írása során előfordulhat, hogy utólag szúrunk be egy ábrát, vagy hogy megváltoztatjuk az ábrák sorrendjét. Ekkor manuálisan frissíteni az ábrasorszámokat és ábrahivatkozásokat nagyon időigényes és unalmas munka, aminek során könnyű hibát véteni és például két darab 5. ábrát hagyni a dokumentumban. Automatikusan generált sorszámok esetén azonban további körültekintést igényel, hogy a határozott névelő használatát elkerüljük a hivatkozott ábrák sorszáma előtt, mert ha a sorszám megváltozik, akkor esetleg a határozott névelőt is meg kell változtatnunk.

Ha egy ábra az oldal aljára kerül, akkor könnyen előfordulhat, hogy az ábrához tartozó képaláírás már a következő oldal tetejére csúszik át, ami az olvasó számára nagyon zavaró tud lenni. Az ábra és a hozzá tartozó aláírás szétválasztását kétféleképpen is megakadályozhatjuk. Az egyik lehetőség az, hogy az ábrát az aláírásával együtt egy, egyetlen cellából álló táblázatba helyezzük el és a táblázat tulajdonságainál megtiltjuk, hogy oldaltörés lehessen egy soron belül. Ezt a Táblázat tulajdonságai ablak Sor fülének „Oldaltörés a soron belül is lehet” opció kikapcsolásával érhetjük el. A másik, talán kényelmesebb lehetőség az az, hogy létrehozunk egy ábra nevű stílust mondjuk a normál stílusból származtatva, ahol is a Formátum, Bekezdés beállításánál bekapcsoljuk az "Együtt a következővel" opciót.

Azért is hasznos a dokumentum logikai struktúráját pontosan követő stílusokat használni, mert ebben az esetben a könnyedén tudunk tartalomjegyzéket létrehozni. A Hivatkozások szalag Tartalomjegyzék csoportjának Tartalom ikonjával tudunk a dokumentumból generált tartalomjegyzéket beszúrni. Alapértelmezésként a címsor stílusokat felhasználva jön létre a tartalomjegyzék, de ezt szükség esetén testreszabhatjuk.

---

1 Érdemes megfigyelni, hogy ugyanarra a fogalomra a Word sajnos három különböző kifejezést használ: felirat, címke és hivatkozástípus.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Ha nem saját eredményeinkről írunk egy szakdolgozatban, akkor meg kell adnunk az adott részhez felhasznált forrásmunka részleteit. Ennek tudományterületenként eltérő szokásai vannak. A Word a legtöbb idézési stílushoz kényelmes keretrendszert biztosít. A Hivatkozás szalag Idézetek és irodalomjegyzék csoportja nyújt ehhez támogatást. A funkciók az ábrahivatkozások ismeretében könnyedén elsajátíthatóak.

3. Formulák használata Excelben

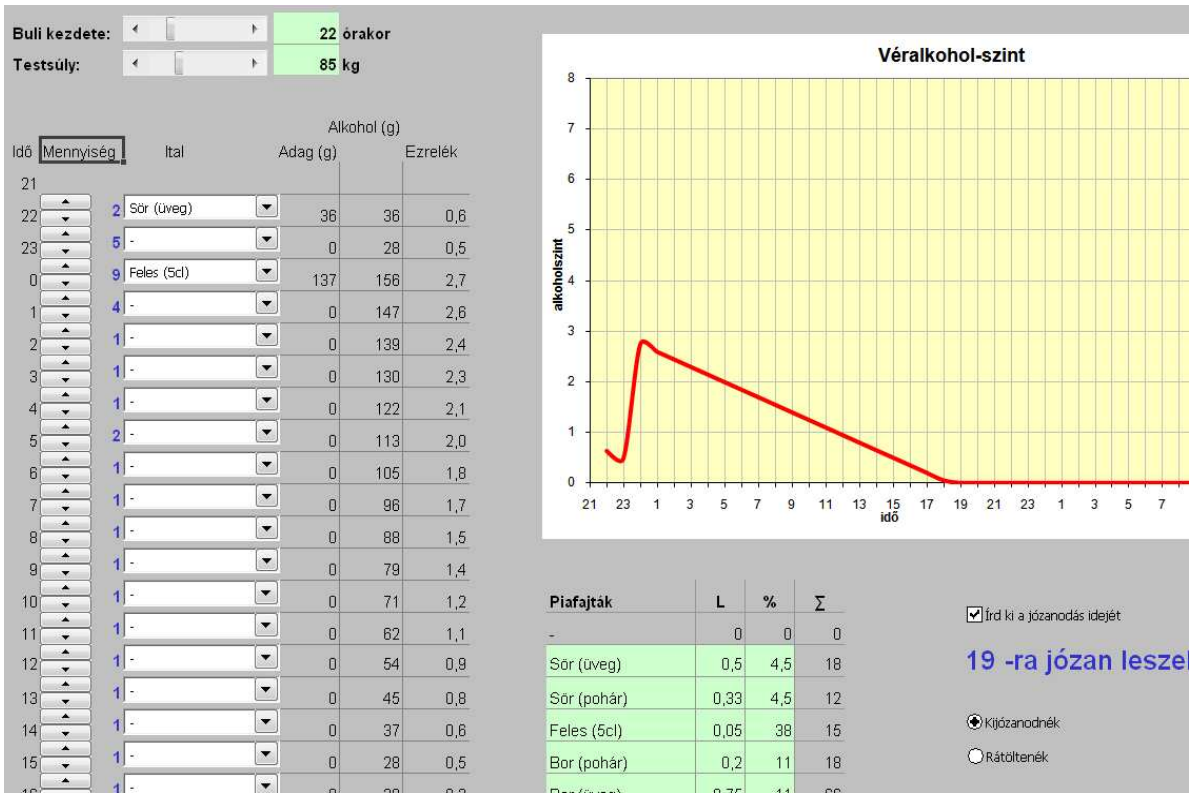
Relatív, abszolút címzés. Munkalap-függvények: szum, ha, fkeres, max, hol.van, index. Képletvizsgálat: elődök, utódok feltérképezése

Az Excel program elsősorban adatok strukturált tárolására és feldolgozására használható. Az adatbevitel remélhetőleg kellően intuitív. Ebben a fejezetben néhány egyszerű adatfeldolgozásra használható formulával és munkalap-függvénnyel ismerkedhetünk meg. Valamint pár adatbevitelt megkönnyítő funkcióval.

Pár évvel ezelőtt a világ leghasznosabb Excel munkafüzete címmel terjedt egy köremail, amiben a 3-1. ábrán látható munkalap volt. Ebben a fejezetben ennek a munkalapnak a reprodukálásán keresztül ismerhetünk meg pár hasznos Excel-funkciót.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



3-1. ábra. Képernyőkép a véralkoholszint-kiszámító munkalapról.

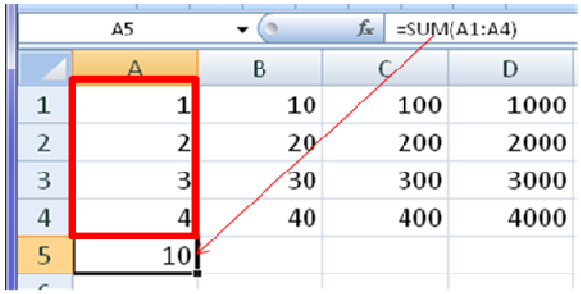
Vegyük észre, hogy a 3-1. ábra bal felső sarkában szerepel a buli kezdetének időpontja és a bal oldali "idő" oszlopban ehhez a kezdési időponthoz igazítva láthatóak a nap órái. Hogyan tudjuk az idő oszlopot úgy létrehozni, hogy ha a munkalap felhasználója megváltoztatja a kezdési időpontot, akkor automatikusan az idő oszlop is megváltozzon?

Mielőtt a fenti kérdésre választ kapnánk, érdemes tudni, hogy a munkalap cellái nem csak számokat és szövegeket tartalmazhatnak, hanem úgynevezett formulákat is vagy más néven képleteket is. Ezeknek a formuláknak az értéke egy számítási művelet eredményeképpen áll elő. Tipikusan egy ilyen formula más cellák tartalmára hivatkozik, az ott található értékekből állítja elő a formulát tartalmazó cella értékét. A 3-2. ábrán, az A5 cellában egy formula értéke látható; maga a formula a jobb felső sorokból olvasható ki. Látható, hogy a formula egyenlőségjellel van bevezetve, a formula pedig egy munkalapfüggvényből áll.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Az angol SUM, a magyar nyelvű Excelben a SZUM függvény a paraméterül kapott tartomány celláinak összegét számolja ki.



3-2. ábra. Egy egyszerű Szum formula

Az az érdekes, hogy ha az A5 cellát a vágólapra helyezzük, mondjuk, a Control-C gyorsbillentyűvel és a B5 cellánál beillesztjük a vágólap tartalmát (Control-V), akkor formula ugyan átmásolódik, de a szumma függvény argumentuma A1:A4-ről B1:B4-re változik. Illetve azt is mondhatnánk, hogy a formulában szereplő tartomány a formulához képesti (relatív) pozíciója nem változik meg. A vágólapra helyezés helyett egyébként az aktív A5-ös cella bal alsó sarkát megfogva és jobbra húzva is lemásolhattuk volna a formulát a B5-ös cellába.

Nézzünk egy másik példát! A 3-3. ábrán két munkalap látszik. Mindkettő B2-es cellájában lévő formula az A2-es és B1-es cella értékét szorozza össze. Az eredmény mindkét esetben 10. Azonban ha a formulát lemásoljuk a B3, B4, B5 cellákba, akkor B4 és B5 cellák esetén eltérő eredményeket kapunk a két munkalap esetén. Ennek az az oka, hogy az egyik esetben a formula a B1-es cellára B$1-ként hivatkozik, emiatt a cella másolásakor a cím sor része változatlan marad abszolút értelemben. Tehát ha a cella címének sor és/vagy sorrésze elé dollár jelent teszünk, akkor a cella másolásakor a cím a munkalaphoz képest nem fog megváltozni. Ha nincs dollár jel, akkor pedig a formulához képest nem fog megváltozni. A 3-3. ábrán látható példa jobboldali részén a szorzatok első tényezője relatív értelemben nem változik, hiszen mindig a formulától balra lévő cellára hivatkozik, míg a szorzat második tényezője abszolút értelemben nem változik, hiszen mindig a B oszlop első sorára hivatkozik.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

	A	B	C
1		10	
2	1	10	=A2*B1
3	2	20	=A3*B2
4	3	60	=A4*B3
5	4	240	=A5*B4

	A	B	C
1		10	
2	1	10	=A2*B$1
3	2	20	=A3*B$1
4	3	30	=A4*B$1
5	4	40	=A5*B$1

3-3. ábra. Relatív és abszolút címzés. A könnyebb érthetőség kedvéért a B oszlop mellett a B oszlopban lévő cella formulája található.

Visszatérve tehát a "világ leghasznosabb" munkalapjára, tegyük fel, hogy az F2 cella tartalmazza a felhasználó által megváltoztatható kezdési időpontot. Első próbálkozásunk során a B7 cellába olyan formulát írunk, ami egyszerűen a kezdési időpontra hivatkozik, az alatta lévő cellában pedig olyan formulát írhatnánk, ami a felette lévő értékhez add hozzá egyet (3-4. ábra). Ezzel a megközelítéssel az a baj, hogy nem veszi figyelembe, hogy egy nap csak 24 órából áll.

	A	B	C
6		idő:	
7		20	=F2
8		21	=B7+1
9		22	=B8+1
10		23	=B9+1
11		24	=B10+1
12		25	=B11+1
13		26	=B12+1
14			

3-4. ábra. Első kísérlet a buli óráinak felsorolására. Az F2 cella tartalmazza a kezdési időpontot.

	B	C	D
6	idő:		
7	20	=F2	
8	21	=HA(B7>=23,0,B7+1)	
9	22	=HA(B8>=23,0,B8+1)	
10	23	=HA(B9>=23,0,B9+1)	
11	0	=HA(B10>=23,0,B10+1)	
12	1	=HA(B11>=23,0,B11+1)	
13	2	=HA(B12>=23,0,B12+1)	

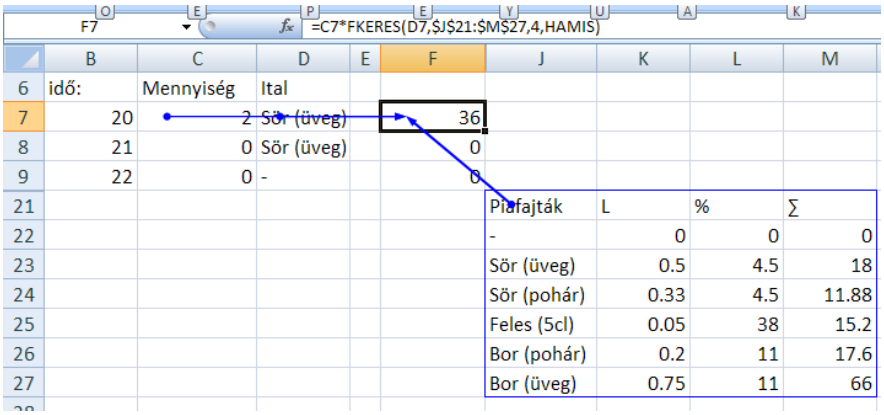
3-5. ábra. A buli óráinak helyes felsorolása.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Első próbálkozásunk hibáját egy kicsit összetettebb formula megírásával tudjuk korrigálni. A "HA" munkalapfüggvény három argumentumot vár, az első egy feltételes kifejezés, a második a függvény értéke abban az esetben, ha a feltételes kifejezés igaz lesz, a harmadik argumentum pedig a függvény értéke abban az esetben, ha a feltételes kifejezés hamis lesz. Második próbálkozásunkban (3-5. ábra) látható, hogy ha az előző cella értéke nagyobb vagy egyenlő volt 23-nál, akkor a formulák értéke 0 lesz, egyébként pedig egy sorral feljebb lévő cellánál eggyel nagyobb.

Haladva tovább a munkalap előállításával, a 3-1. ábrán látható, hogy az ital típusa melletti oszlopban az adott órában fogyasztott italok összes alkoholtartalma található. Hogyan lehet ezt kiszámítani? Ha tudjuk, hogy egy adott típusú ital (piafajta) hány gramm alkoholt tartalmaz, akkor ennek és az adott órában bevitt mennyiségnek a szorzataként előáll a keresett érték. A nehézséget itt az jelenti, hogy a munkalap felhasználója csak azt adja meg, hogy az adott órában mit fogyasztott, azt nem, hogy ennek a típusú italnak mi az alkohol tartalma. Szerencsére azonban ez a J21 cellánál kezdődő táblázatból kikereshető az FKERES (angolul VFIND) munkalapfüggvény segítségével. A függvénynek négy argumentuma van: az első tartalmazza azt az értéket, amit keresni szeretnénk a második argumentumban megadott táblázat első oszlopában, a harmadik argumentum azt adja meg, hogy keresett érték (esetünkben a D7 cellában található "Sör (üveg)" szöveghez) sorában hányadik cella értéke legyen az FKERES függvény értéke. Ha pedig az utolsó argumentum hamis, akkor pontos egyezés szükséges az első oszlopban történő keresés során. Ezen a ponton azonban érdemes a beépített Sugó használatát elsajátítani: nézzük meg mit ír a Súgó az FKERES működéséről, ha annak a negyedik argumentuma nem hamis.



	B	C	D	E	F	J	K	L	M
6	idő:	Mennyiség	Ital						
7	20	2	Sör (üveg)		36				
8	21	0	Sör (üveg)		0				
9	22	0	-		0				
21						Piafajták	L	%	Σ
22						-	0	0	0
23						Sör (üveg)	0.5	4.5	18
24						Sör (pohár)	0.33	4.5	11.88
25						Feles (5cl)	0.05	38	15.2
26						Bor (pohár)	0.2	11	17.6
27						Bor (üveg)	0.75	11	66

3-6. ábra. Példa az FKERES munkalapfüggvény használatára.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

A G oszlopban nem az adott órában a szervezetbe bevitt alkohol mennyisége található, hanem az összes szervezetben lévő alkohol mennyisége, tehát az is, amit a felhasználó korábban fogyasztott és a szervezete még nem bontott le. Használjuk azt az egyszerű modellt, hogy a lebontási sebesség 0,1 gramm óránként és testsúly-kilogrammonként. Feltételezve, hogy a felhasználó a buli előtt nem fogyasztott alkoholt, az első óra végén a szervezetében lévő összalkohol megegyezik az első órában bevitt mennyiséggel, tehát G7 formulája "=F7". A későbbi órákban az adott órában bevitt mennyiséghez kell hozzáadni a korábbi órából megmaradt (el nem bontott) mennyiséget. Így tehát például G8 cella formulája "=F8+MAX(0, G7-0.1 * F3)", ha feltételezzünk, hogy a felhasználó testsúlyát az F3-as cellában adta meg. Érdemes megfigyelni, hogy a MAX munkalapfüggvény segítségével megakadályoztuk, hogy a szervezetben bent lévő alkoholmennyiségnél többet bontson le a szervezet. Továbbá abszolút címzést használtunk a testsúlyra való hivatkozáskor azért, hogy a formula másolás után is helyes maradjon.

Ahhoz, hogy megállapítsuk a kijózanodási, illetve a rátöltési időt a HOL.VAN és az INDEX munkalap-függvényekre van szükségünk. A HOL.VAN egy megadott tartományban megkeres egy értéket és annak a sorszámát adja vissza. Az INDEX segítségével egy adott tartomány (tömb) elemeit lehet megcímezni. A pontos használatért megint csak érdemes a Súgót áttanulmányozni, mert fontos, hogy a Súgó helyes használatát is elsajátítsuk. A kijózanodási idő megtalálásának első lépéseként (3-7. ábra) keressük meg azt a sort a munkafüzetben, amihez tartozó, a felhasználó szervezetében lévő véralkoholszint nulla lesz. Ezt a =HOL.VAN(0, G7:G20, 0) formulával tehetjük meg. Az ábrán látható példában a formula értéke 11 lesz, ami azt jelenti, hogy a G7:G20 tartomány 11. cellája nulla először. A tartomány 11. cellája a G17-es cella. Második lépésben az érdekel minket, hogy a G17-es cella a buli hányadik óráját jelöli, tehát az, hogy a B7:B20-as tartomány 11. cellájának mi az értéke. Ezt az =INDEX(B7:B20, 11) formulával tudjuk megkapni. Általánosabb lesz a megoldásunk, ha a második formula az első eredményét használja fel ahogy az a 3-7. ábrán is látszik. Azonban semmi nem akadályoz meg minket abban, hogy magát a HOL.VAN függvényt az INDEX-be ágyazzuk így: " =INDEX(B7:B20, HOL.VAN(0,G7:G20,0))". Jól lehet, az ilyen egymásba ágyazások során nem szükséges a részeredményeket segédcellában letárolni (mint például a HOL.VAN értékét G23-ba), a nagyon hosszú és összetett formulákat inkább érdemesebb több részre bontani. A több részre bontott formulák jobban átlátható kódot eredményeznek, amiben szükség esetén könnyebb hibákat keresni.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

	B	G	H	I
5		Alkohol (g)		
6	idő:		Ezrelék	
7	20	36	0.54	
8	21	44	0.66	
9	22	34	0.51	
14	3	29.6	0.444	
15	4	19.6	0.294	
16	5	9.6	0.144	
17	6	0	0	
18	7	0	0	
19	8	0	0	
20	9	0	0	
21				
22				
23		11	=HOL.VAN(0,G7:G20)	
24		6	=INDEX(B7:B20,G23)	

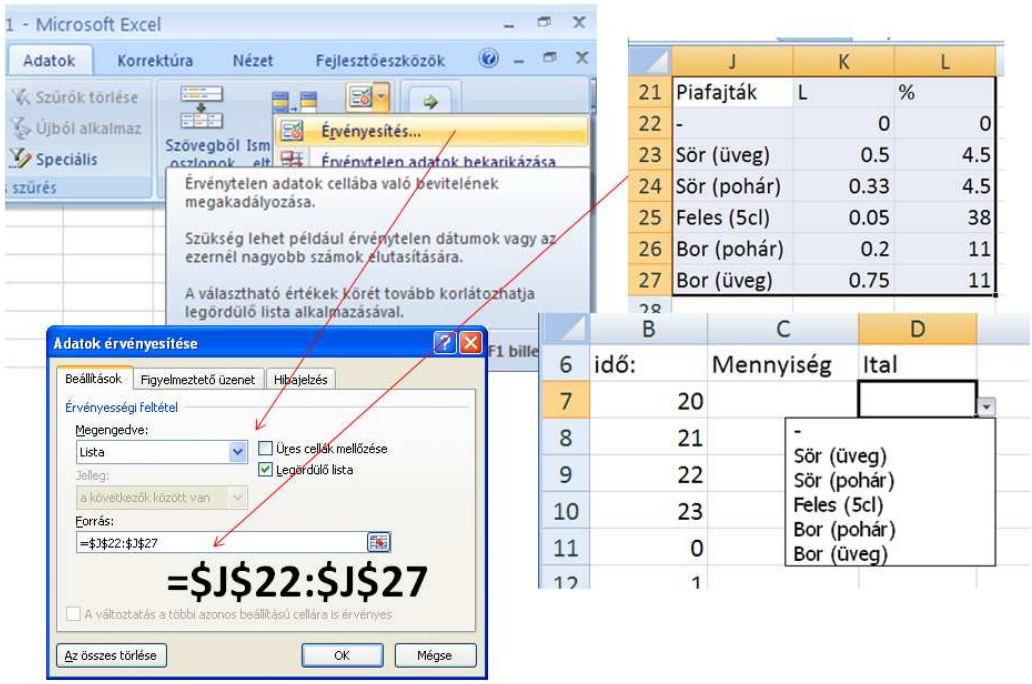
3-7. ábra. Példa a HOL.VAN és az INDEX munkalapfüggvények használatára

4. Űrlapok készítése Excelben

Adatérvényesség-ellenőrzés. Űrlap-vezérlőelemek használata, cellához való kötésük. Lapvédelem, munkalap zárolása. Grafikonok.

A 3-1. ábrán látható, reprodukálni kívánt munkalapon minden órában egy legördülő menüben megadhatja a felhasználó, hogy milyen típusú italt fogyasztott. Ezt többféleképpen érhetjük el, de legkényelmesebb módja az ilyen legördülő menüből elérhető választára az adatérvényesség-ellenőrzésének használata (4-1. ábra). Ha a D oszlopban adható meg az adott órában elfogyasztott ital típusa, akkor azokat a cellákat előzetesen kiválasztva az Adatok szalagon az Adateszközök csoport Érvényesítés ikonján keresztül tudjuk megadni, hogy az adott cellákba milyen értékek fordulhatnak elő. Nekünk most csak a listában felsorolt érvényességi feltételt kell használni, de érdemes a többi feltételt is áttanulmányozni. A lista forrásaként adjuk meg a J22:J27 tartományt, mert ott vannak az italok felsorolva és már készen is vagyunk. Érdemes észrevenni, hogy a forrás tartománya abszolút címzéssel került bejegyzésre, így a D7 cella másolása során érvényesítési feltétel is helyesen fog lemásolódni.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu



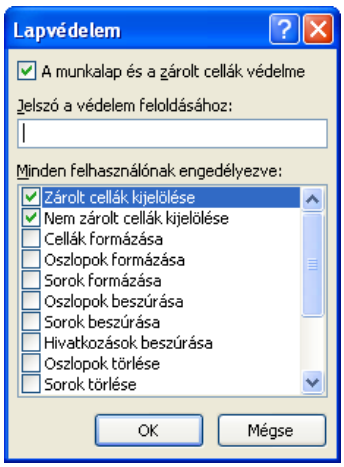
4-1. ábra. Adatérvényesség ellenőrzésének beállítási lépései.

A 3-1. ábrán a buli kezdetét, a felhasználó testsúlyát és az adott órában elfogyasztott italok mennyiségét csúszkák vagy más néven görgetősávok segítségével is meg lehet adni. Ezek a csúszkák jó példák az úgynevezett űrlap-vezérlőelemekre. Ilyen vezérlőelemeket csak akkor tudunk beszúrni, ha előtte az Excelben bekapcsoljuk a Fejlesztőeszközök szalagot. Nyomjuk meg a bal felső sarokban lévő Office gombot, majd "az Excel beállításai" után a kapcsoljuk be a "Fejlesztőeszközök lap megjelenítése a szalagon" opciót. Ezen az újonnan megjelent szalagon Beszúrás ikon megnyomása után kiválasztatjuk a beszúrni kívánt vezérlőelemet. Akár csuszkát, jelölőnégyzetet, választógombot, vagy bármilyen más vezérlőelemet használunk egy dolog közös a használatunkban: meg kell adnunk, hogy a vezérlőelem mit vezérel, azaz egy csatolást kell létrehoznunk egy cella és a vezérlőelem között. A vezérlőelem jobbklikkel elérhető felugró menüjének Vezérlőelem formázása menüpontjával tudjuk a csatolást beállítani. A csatolás beállítása után a vezérlőelem változtatásával együtt változik a csatolt cella értéke, illetve a cella változtatásával megváltozik a vezérlőelem állapota is.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Ha az általunk összeállított munkafüzetet másoknak is oda kívánjuk adni, és szeretnénk, hogy a munkalapot csak az általunk előzetesen meghatározott helyen módosítsák, akkor érdemes a lapvédelmet használnunk. A lapvédelmet a Korrektúra szalag Változtatások csoportjából tudjuk bekapcsolni. A lapvédelmi beállításokat a 4-2. ábrán láthatjuk. Ha bekapcsoljuk a lapvédelmet, akkor az alapértelmezett beállítások mellett a zárolt cellák tartalmát nem tudjuk módosítani. Viszont eredetileg minden cella zárolt. Egy cella vagy egy tartomány zárolási tulajdonságát a kijelölt tartomány jobbklikkel előhozható környezetfüggő menüjéből állíthatjuk be (4-3. ábra).

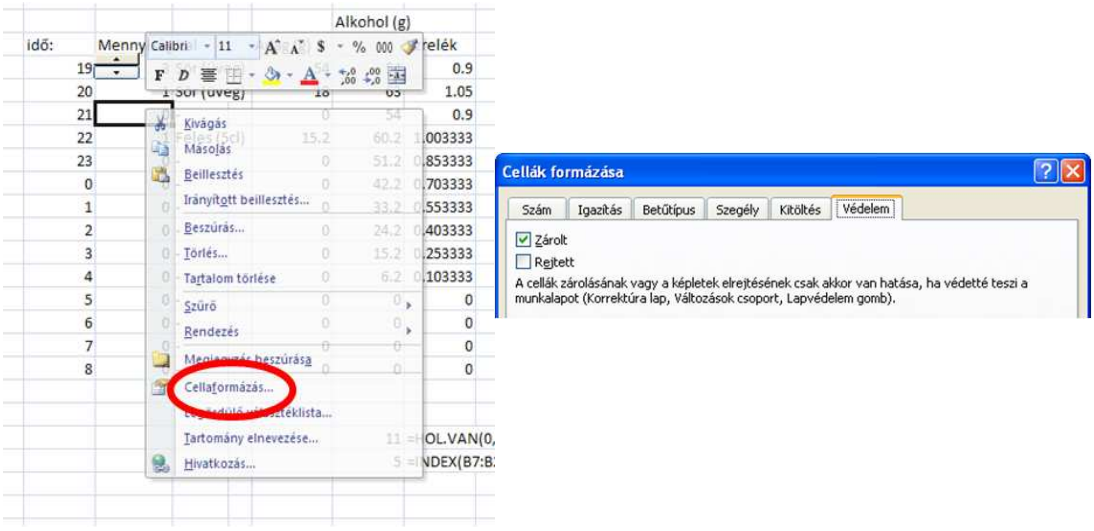


4-2. ábra. Lapvédelem beállítási lehetőségei

A lapvédelem és a megfelelő adatérvényesség-ellenőrzés együttes használata nagyon előnyös tud lenni. A két funkció együttes használatával olyan munkafüzeteket tudunk kialakítani, amivel pontosan meg tudjuk határozni, hogy milyen cellába milyen adatokat írhatnak be a munkafüzetünk felhasználói.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



4-3. ábra. A zárolás tulajdonság módosítása

A felhasználó véralkoholszintjének időbeli változását egy grafikonon is nyomon követhetjük. Grafikon beillesztésére a Beszúrás szalagon van lehetőségünk. A grafikon beillesztése ugyan nem bonyolult, de érdemes odafigyelni, hogy az adatsor időtengelyén a tényleges időértékeket helyesen állítsuk be.

5. Körlevél

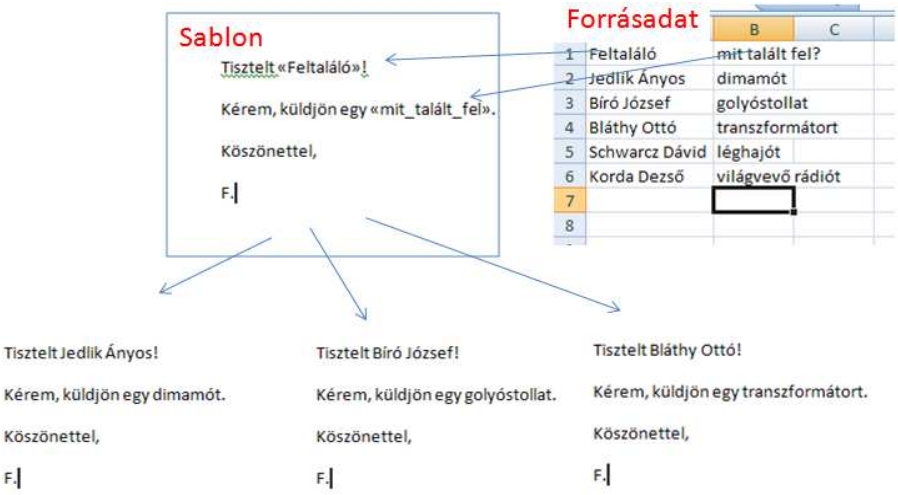
Körlevél készítése Excel munkalapon tárolt forrásadatokból. Egyéni adatmezők használata

A Word dokumentumszerkesztővel lehetőségünk van klasszikus értelemben vett körlevelek írására, azonban a Word körlevél-készítési képességeire érdemesebb egy kicsit általánosabban nézni. Így számos, unalmasan ismétlődő feladat elvégzését tudjuk majd elkerülni. A körlevél funkciót nem csak levelek elkészítésére használhatjuk, hanem tetszőleges hasonló tartalmú dokumentumhalmaz legenerálására is. Körlevél készítése során egy dokumentumsablon segítségével állíthatunk elő több dokumentumot úgy, hogy a sablonban lévő változó adatmezők egy adatforrásból helyettesítődnek be. Maga az adatforrás lehet egy távoli adatbázis lekérdezés eredménye is, de mi leggyakrabban egy Excel munkafüzet munkalapját fogjuk használni.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Az 5-1. ábra egy egyszerű példát mutat be. Fontos, hogy a forrásadatoknál az Excel táblázatban egy sor egy dokumentum változó adatait tartalmazza, azaz egy sor cellái egy-egy adatmező-értéket tartalmazzanak. Továbbá az adattáblázatnak kell, hogy legyen fejléce, azaz az első sor az adatmezők neveit kell, hogy tartalmazza. A példában tehát az A oszlop a feltalálókat tartalmazza, a B oszlop pedig a feltaláló találmányának nevét. A dokumentumsablonban a dokumentumonként eltérő részt az adatmezők nevei adják meg.



5-1. ábra. Körlevél létrehozása egy dokumentumsablon és egy Excel táblázat segítségével

Legegyszerűbben a Levelezés szalag "Körlevélkészítés indítása" ikonjára való klikkeléssel tudunk nekifogni a fenti példa előállításának. A ikonra való klikkelés után felugró menü "Lépésenkénti körlevélvarázsló" menüelemét kell kiválasztanunk. Első lépésként a körlevél típusát kell megadni. Ha általános Word dokumentumot szeretnénk előállítani, akkor a levél típust érdemes választani. A második lépésben a dokumentumsablont lehet megadni. Ha nincs előkészített anyagunk, akkor nyugodtan adjuk meg az (üres) aktuális dokumentumot, mert lesz még lehetőségünk a sablonként szolgáló aktuális dokumentum szerkesztésére. A következő, harmadik lépésben a címzettlistát kell megadnunk. Itt adjuk meg a fenti szabályok szerint létrehozott Excel fájlt. A negyedik lépésben a sablont tudjuk szerkeszteni. Ebben a lépésben lehet a sablonba az adatmezőket beszúrni. A további elemek menüpontnál tudjuk kiválasztani azt az adatmezőt, amit a kurzor aktuális pozíciójához be kívánunk szúrni. Abban az esetben, ha az előre definiált mezőneveket szeretnénk használni, de a saját adatbázisunkban más oszlopneveket használtunk, akkor jól jöhet a "Adatmezők beszúrása" ablak alján

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

található "Mezők egyeztetése…" nyomógomb, ami segítségével az adatbázisban található és a körlevélkészítő által használt mezőneveket tudunk megfeleltetni egymásnak. Az ötödik lépkor a dokumentumok a sablon alapján legenerálódnak; itt van lehetőségünk megnézni a dokumentumgenerálás eredményét. Amennyiben nincs módosításra szükség, akkor a varázsló utolsó, hatodik pontjával véglegesíthetjük a körlevél készítését az egyesítés befejezésével.

Tipikus irodai munkafolyamatok során a körlevélkészítési funkciót használhatjuk például alkalmazotti szerződések készítésére is. Általában ezek a szerződések csak az alkalmazottak adatiban térnek el, ezeket ezért összegyűjthetjük egyetlen Excel táblázatban és egy előre elkészített szerződéssablon segítségével elkészíthetjük egy új alkalmazott szerződését. Ez azért is előnyösebb egy korábban elmetett szerződés módosításánál, mert a módosítás során könnyen elfeledkezhetünk egy mező cseréjéről, vagy mert az alkalmazott nevét az Excel táblázatban csak egyszer kell megadni, a kész szerződésben pedig általában sokszor előfordul.

6. Kimutatások

Kimutatás készítése (pivot table, chart), frissítése. Forrásadatok helyes megválasztása. Szűrési feltételek. Származékos értékek számítása. Megjelenítés testreszabása.

Excel munkalapokon különböző struktúrákban tárolhatunk adatokat. Fontos, hogy úgy válasszuk meg adatok struktúráját, elrendezését, hogy az könnyen átlátható legyen, könnyen bővíthető legyen, de talán legalább ilyen fontos, hogy az adatokból könnyen tudjunk következtetéseket levonni. Az adatelemzésnek gyors és kényelmes módja az úgynevezett kimutatás vagy más néven pivot table készítése. Ennek segítségével ugyanabból az adathalmazból különböző szempontok szerinti kimutatásokat tudunk egyszerűen készíteni úgy, hogy különböző kérdéseket tudunk ugyannak az adathalmaznak a segítségével megválaszolni.

Lássunk egy példát! A 6-1. ábrán egy kis táblázatot látunk, amelynek egy sora azt tartalmazza, hogy az adott keresztnevet az adott évben hány újszülött kapta. Tegyük fel, hogy a táblázatunk 2008 és 2010 között az összes újszülöttről tartalmaz adatokat. Különösen fontos a kimutatások készítésekor, hogy a táblázat fejléces legyen, azaz az első sora tartalmazza az oszlopok neveit.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

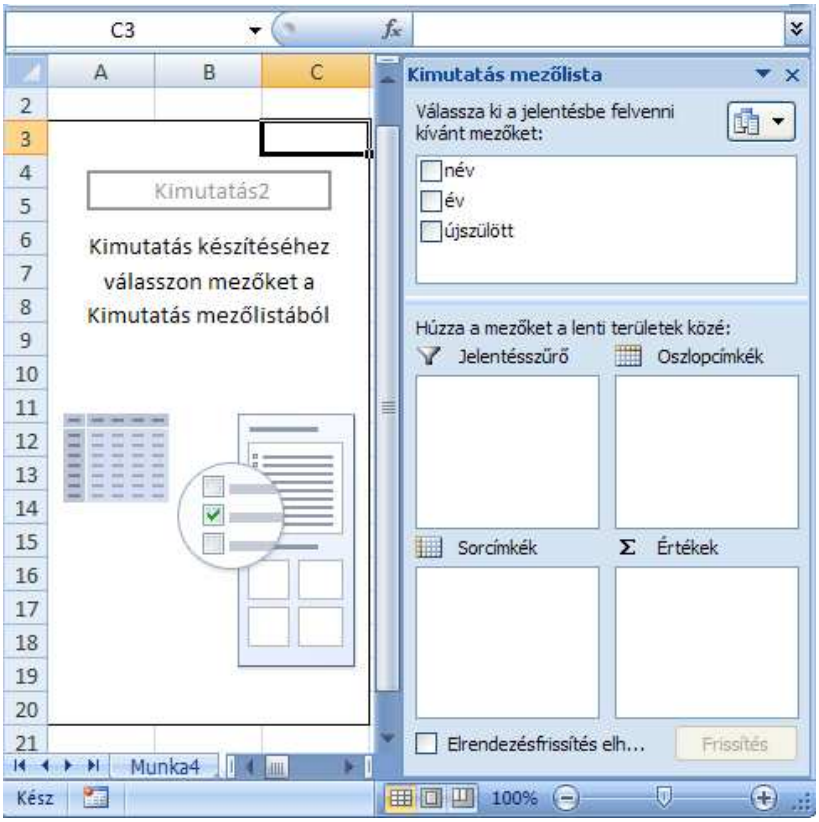
	név	év	újszülött
2	ZSÓFIA	2008	966
3	PETRA	2008	893
4	ZSÓFIA	2009	900
5	PETRA	2009	883
6	ZSÓFIA	2010	804
7	PETRA	2010	783
8			

6-1. ábra. Névadatások gyakorisága (minta)

Az első kérdés, amire választ keresünk ekkor az, hogy csökkent-e az újszülöttek száma az évek során. Az eredeti táblázatunkból ezt nem könnyű megállapítani, mert például nem lehet egyszerűen az adatokat az "újszülött" vagy az "év" oszlop szerint rendezni, hiszen az egy évben születettek adatai több sorban vannak szétszórva. A kimutatás készítéséhez jelöljük ki az A1-es cellát és a Beszúrás szalagon a Táblázatok csoportban klikkeljünk a Kimutatás ikonra, ekkor meg tudjuk adni forrásadat táblázatát illetve tartományát. Mivel azonban mi már előzetesen kijelöltük az A1-es cellát, ezért jelen esetben automatikusan megtörtént forrástartomány beállítása. Így haladhatunk tovább az OK gombbal, aminek eredményeként a 6-2. ábrán látható üres kimutatás fog létrejönni egy új munkalapon. Figyeljük meg a jobb oldalon található Kimutatás mezőlista ablakát! Felül láthatóak a 6-1. ábra táblázatához tartozó oszlopnevek. Ezeket a mezőneveket az egérrel négy különböző négyzetbe húzhatjuk bele: a jelentésszűrőbe, az oszlopcímkékbe, a sorcímkékbe és az értékekbe.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



6-2. ábra. Üres kimutatás-szerkesztő

A 6-3. ábrán az látható, amikor a sorcímkék között az "év" mező szerepel az értékek között pedig az "újszülött" mező. Ezzel tulajdonképpen meg is válaszoltuk a kérdés: az adatok szerint egyre kevesebb gyermek születik. Azonban ne rohanjuk tovább. Hogyan is jutottuk erre az eredményre? A kimutatással a forrásadatok egy tömörebb változatát állítottuk elő, amiből a kérdése adott válasz könnyen leolvasható volt. Az "év" sorcímként való megadásával azt értük el, hogy a forrásadatban előforduló minden különböző év értéknek különböző sora lesz a kimutatásban. De mivel egy évhez több sor is tartozik, ezért az érték doboz határozza meg, hogy az azonos évhez tartozó sorokat hogyan vonjuk össze egyetlen sorrá a kimutatásban. Például 2008-ban 966 Zsófia született és 893 Petra, tehát a 6-1. ábra táblázatában két olyan sor is van, amit valahogy a 6-3. ábra kimutatásának 4. sorába kéne belegyömöszölni. Mivel azonban az érék dobozban "Összeg/újszülött" szerepel ezért a kimutatás B4-es cellájában 966+893 azaz 1859 lesz.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

	A	B
2		
3	**Sorcímkék**	**Összeg / újszülött**
4	2008	1859
5	2009	1783
6	2010	1587
7	**Végösszeg**	**5229**

6-3. ábra. Újszülöttek száma évenkénti bontásban

A második kérdésként arra vagyunk kíváncsiak, hogy az adott időszakban összesen hány gyermek kapta a Petra illetve a Zsófia keresztneveket. Ha most a változatosság kedvéért a sorcímkéket üresen hagyjuk és az oszlopcímkék közé a "név" mezőt vesszük fel, valamint az érték dobozban továbbra is az "Összeg/újszülött"-et tesszük, akkor a 6-4. ábrán látható kimutatást kapjuk. Az eredményt már könnyebben meg tudjuk magyarázni: a kimutatás 4. sora a lehetséges oszlopcímkéket tartalmazza, tehát forrásadatokban előforduló keresztneveket. Mivel Petra a harmadik, ötödik és a hetedik sorban fordul elő, ezért ezek a sorok vannak együttesen ábrázolva a B5 cellában, pontosabban az "újszülött" értékek összege szerepel, mert ezt adtuk meg az érték dobozban.

	A	B	C	D
2				
3		**Oszlopcímkék**		
4		**PETRA**	**ZSÓFIA**	**Végösszeg**
5	**Összeg / újszülött**	**2559**	**2670**	**5229**

6-4. ábra. Keresztnevek népszerűsége a 2008-2010-es időszakban

Összefoglalásként elmondhatjuk, hogy egy kimutatás készítésekor az általunk megadott feltételekre illeszkedő sorokat kereshetjük ki, és ezeken a sorokon számításokat hajthatunk végre, hogy együttesen lehessen ábrázolni az illeszkedő sorokat.

A kimutatás-készítés hasznosságára most nézzünk egy fokkal összetettebb példát. A 6-5. ábra egy képzeletbeli kisvállalkozás egyszerű könyvelését mutatja be. Az utazó ügynök alkalmazottak számláinak felel meg egy-egy sor a táblázatban. A név oszlop az ügynök nevét tartalmazza; a hónap oszlop azt, hogy a számla az év hányadik hónapjára vonatkozik; a típus azt mondja meg, hogy a számla milyen fő kategóriába sorolható; az érték oszlop a számla értékét forintban; a város a számla kiállítási helyét; végezetül a nap oszlop azt adja meg, hogy hotelszámla esetén hány napot töltött az ügynök az adott hotelben. Mivel egy sor egy számlának felel meg, ezért viszonylag könnyű a táblázatot karban tartani.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

	A	B	C	D	E	F
1	név	hónap	típus	érték	város	nap
2	Gáz Géza	1	étel	8 000	Pécs	
3	Gipsz Jakab	1	benzin	9 500	Szeged	
4	John Doe	1	szállás	60 000	Debrecen	1
5	Gáz Géza	2	étel	2 500	Pécs	
6	Gáz Géza	3	benzin	7 000	Pécs	
7	Gáz Géza	2	szállás	120 000	Pécs	1
8	Gipsz Jakab	6	szállás	160 000	Szeged	3
9	Gipsz Jakab	2	benzin	11 500	Szeged	
10	John Doe	4	étel	5 000	Debrecen	
11	John Doe	3	étel	16 000	Szeged	
12	John Doe	2	étel	700	Pécs	

6-5. ábra. Kiküldetéssel kapcsolatos mintaadatok

Első feladatunk legyen típus szerint bontásban megadni, hogy ki mennyit költött. A kérdés megválaszolható egy olyan kimutatással, aminél a sormezőknél a "név" szerepel, az oszlopmezőknél a "típus", az értékeknél pedig az "Összeg/érték". Ez utóbbinál a mezőkód legördülő menüjénél értékmező-beállításoknál van lehetőségünk megadni, hogy az oszlop- és sorértékre illeszkedő sorokat a kimutatás ne megszámolja, hanem az értékek összegét képezze.

Második feladatként adjuk meg a havi átlagos tankolásra költött összeget, valamint az adott hónapban az egy alkalommal kifizetett legdrágább tankolást is. Mindezt úgy, hogy az eredmény a havi átlagok csökkenő sorrendjében legyen. Mivel ennél a feladatnál csak a benzinszámlák érdekelnek minket a Jelentésszűrő dobozhoz adjuk hozzá a típus mezőt és a kimutatás tetején állítsuk be a benzin szűrési feltételt. Majd a sorcímkék közé húzzuk be a hónap mezőnevet. Ezek után a havi átlagos tankolásra költött összeget nem nehéz megkapni, hiszen csak az érték mezőnevet kell az Értékekhez hozzáadni, valamint a mezőnév legördülő menüjében kell az értékmező-beállításoknál az átlagot kikeresni. A korábbi feladatoktól eltérően most még egy származtatott mennyiségre is kíváncsiak vagyunk: a havi legdrágább tankolásra is. Nincs más dolgunk, mint megint hozzáadni az érték mezőnevet az Értékek közé, most azonban az átlag helyett a maximum függvényt kell kiválasztanunk. Végezetül a kimutatás sorcímkék cellájának legördülő menüjében van lehetőségünk egyéni rendezési feltételeket megadni. Itt a további rendezési lehetőségek között tudjuk kiválasztani az átlag/érték oszlopot.

Harmadik feladatként nézzük meg, hogy Debrecenhez képest százalékosan mennyivel költöttek többet ételre Pécset és Szegeden. Nyilván most is az előző feladathoz hasonlóan érdemes a Jelentésszűrőbe a

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

típust tenni, de most az ételt kell kiválasztani. Továbbá az Oszlopcímkék közé tenni a várost és az értéknél az "Összeg/érték"-et kell beállítani. Az újdonság a korábbi feladatokhoz képest abból adódik, hogy az eredményeket egy másik származtatott mennyiséghez képest szeretnék ábrázolni. Ezt az értékmező-beállítások módosításával tudjuk elérni: az értékek megjelenítése fülön válasszuk a százalékos eltérést, majd viszonyítási mezőt állítsuk "város"-ra és a viszonyítási tételt "Debrecen"-re. (Jelen feladatnál a sorcímkék használata azért nem jó, mert viszonyítási mezőket csak oszlopokhoz lehet megadni.)

Végezetül a kiküldetés hosszának függvényében adjuk meg az egy napra vetített szállásdíj átlagos értékét. Ez a feladat azért nehéz, mert a kimutatás készítése során nincs lehetőségünk az egy napra vetített szállásdíjat kiszámolni. Ezért az egyik lehetőségünk készíteni egy kimutatást, amiben a kiküldetés hosszának függvényében megadjuk meg a szállásdíjak átlagos értékét, ezután a kimutatás mellett létrehozunk egy új oszlopot, amiben elosztjuk a szállásdíj átlagos értékét a kiküldetés hosszával. A másik általánosabban használható megoldás az, hogy a forrásadatokat bővítjük még egy új G oszloppal. A G oszlopba egy olyan formulát írunk, ami a sorhoz tartozó D/F értéket számolja ki. Ezek után az új oszloppal a feladatot megoldó kimutatás már könnyen elkészíthető. Ez a feladat arra volt példa, hogy a kimutatások ugyan nagyon sok mindenre használhatóak, mégis érdemes mindig elgondolkozni, hogy az adott feladatunk megválaszolásához a forrásadatok megfelelő alakra vannak-e már hozva.

Ebben a fejezetben a kimutatás-készítés számos lehetőségét vettük sorra, de nem tértünk ki az összes beállítási lehetőségre, ezért érdemes a különböző beállítási lehetőségeket egyszer végigbogarászni, hátha egy későbbi probléma megoldása során jól jön majd a megismert funkció.

7. További munkalap-függvények

Darab, ha, nagy, átlag. Képletek szerkesztése több lépcsőben. A súgó használata.

Tegyük fel, hogy egy tantárgy oktatása tizenhárom gyakorlatból és egy úgynevezett ellenőrző mérésből áll. Félévvégi jegyet csak az kaphat, aki legalább a gyakorlatok 70 százalékát legalább elégséges osztályzattal teljesítette és átment a kötelező ellenőrző mérésen is. Ezek után a hallgató féléves jegye a kilenc legjobb órai gyakorlatra kapott jegy és az ellenőrző mérésre kapott jegy kerekített átlaga, amiben az ellenőrző mérés jegye 50 százalékos súllyal szerepel. A megoldandó feladatunk kiszámolni egy osztály

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

illetve egy évfolyam hallgatóinak félévvégi jegyét. A gyakorlatokra adott jegyek a 7-1. ábrához hasonlatosan vannak megadva.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Név	1	2	3	4	5	6	7	8	9	10	11	12	13	ellenőrző mérés	
2	Bárány Róbert	5	5		5		3	3	3	3	4	5	5	1	3	
3	Békésy György	5	4	2	5		5	5	5	5	3	5	3		5	
4	Gábor Dénes	5	4	5				4	5	5	5	3	5	2	5	
5	Harsányi János	5	3	3	4	5	4	4	5	5	3	4	4	3	2	
6	Hevesy György	5	2	5			4	4		2		4		4		
7	Kertész Imre	5		4	5	4	3	5	2	5	5	2	5	3	1	
8	Lénárd Fülöp	5		5						4			4		5	
9	Oláh György	5		2		4	4	4	4	4	4	4	5	4		
10	Polányi János	5	5	1		5	3	1	3	1	3	1	5	5		
11	Szent-Györgyi Albert	5		4												
12	Wigner Jenő	5	5	5	5	5	5	5	5	5	5	5	5	5	5	
13	Zsigmondy Richárd		5	5		5	5	5	5		5	5		3	2	

7-1. ábra. Mintaadatok a félévvégi jegy kiszámításához

Ez a feladat elsőre bonyolultnak tűnhet és majd látni is fogjuk, hogy tényleg igen összetett formulákat kell megalkotunk, azonban kisebb részfeladatokra bontva a feladatot mindig könnyen érthető lépésekben fogunk a célunkhoz közelíteni. Ez a probléma-dekompozíciós elv nem csak most, de általánosságban is nagyon hasznos.

A feladat megoldása előtt azonban tekintsük át pár munkalapfüggvény használatát, amik a problémamegoldás során hasznosak lesznek. A "DARAB" függvény megszámolja, hogy az argumentumában átadott tartományok hány darab számot tartalmaznak. A "HA" munkalapfüggvény három argumentumot vár, az első egy feltételes kifejezés, a második a függvény értéke abban az esetben, ha a feltételes kifejezés igaz lesz, a harmadik argumentum pedig a függvény értéke abban az esetben, ha a feltételes kifejezés hamis lesz. A "NAGY" függvény szintaxisa a következő: NAGY(tömb, k), visszatérési értéke pedig a tömb (egy egydimenziós tartomány) k-adik legnagyobb eleme. Végezetül az "ÁTLAG" munkalapfüggvény az argumentumai átlagát számolja ki.

Az előző bekezdés leírása túlságosan tömörnek tűnhet. Ha annak alapján nem lenne világos az egyik vagy másik munkalapfüggvény működése, nézzünk utána a Súgóban a részletes leírásának. Általában is érdemes elsajátítani a beépített Súgó használatát, mert egyrészt az a későbbi munkánk során mindig kéznél lesz, másrészt pedig nincs lehetőség és nem is érdemes minden munkalapfüggvény működését

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

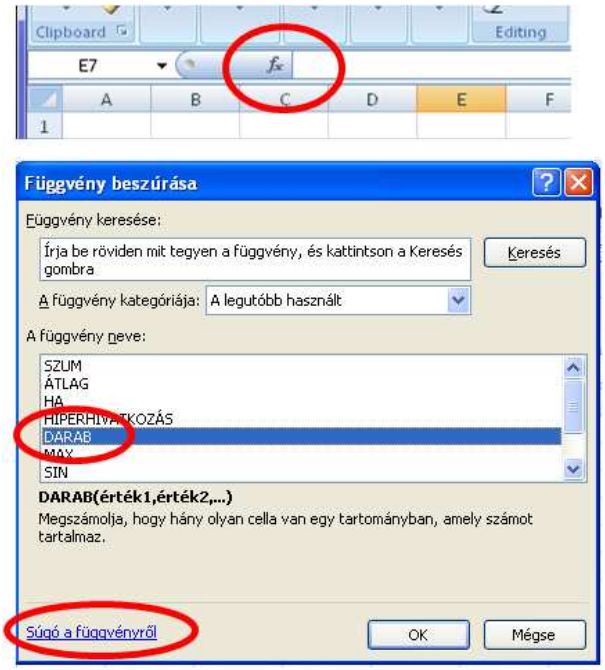
előre részletesen elsajátítani. Elegendő, ha megtanuljuk, hogy hogyan tudjuk megkeresni az aktuális feladatunk megoldásához szükséges munkalap-függvényeket, és hogy hogyan tudjuk ezen függvények pontos működését megérteni. Ahogy az a 7-2. ábra felső részén is látható, a munkalap felet található az "fx" szimbólum, amivel a Függvény beszúrása ablakot tudjuk előhívni. A Függvény beszúrása ablakban egyrészt kategóriák szerint tallózhatunk az elérhető függvények között, vagy szöveges kereséssel kísérelhetjük meg megtalálni a használni kívánt munkalap-függvényünket. Ha az ablak közepén található a listából kiválasztunk egy elemet, például az ábrán látható DARAB-ot, akkor az ablak alján egy rövid leírást kapunk a függvény működéséről. Ez többnyire elegendő akkor, ha már használtuk korábban ezt a munkalapfüggvényt, vagy tisztában vagyunk a működésével, csak a szintaxisára nem emlékszünk pontosan. Ebben az esetben ez a tömör emlékeztető általában elegendő, hogy helyesen tudjuk használni a függvényt. Ha azonban ez az emlékeztető mégsem elegendő nekünk, akkor az ablak bal alsó sarkában lévő linken keresztül tudunk részletesebb leírást kérni a kiválasztott függvényről.

Mindezek után térjünk vissza a fejezet feladatára. Apró lépésekben haladva számoljuk ki a hallgatók félévvégi jegyét. Elsőre adjuk meg a P oszlopban, hogy egy hallgató hány gyakorlaton vett részt. Ha egy hallgató részt vett egy gyakorlaton, akkor arra osztályzatot kapott. Ha rosszul teljesített, akkor elégtelent kapott, de mindenesetre kapott egy jegyet. Így ha megszámoljuk, hogy hány számot tartalmazó cella van a hallgató gyakorlati jegyeit tartalmazó tartományban, akkor megkaphatjuk, hogy az illető hány gyakorlaton vett részt. Tehát P2 cella formulája legyen ez: "=DARAB(B2:N2)". A cella jobb alsó sarkát lehúzva pedig a teljes évfolyamra ki tudjuk terjeszteni a számításunkat.

Második részfeladatként a következő, Q oszlopba adjuk meg, hogy a hallgatók az órák hány százalékán adottak le megoldást, azaz az órák hány százalékán voltak jelen. Szerencsére most nem kell nulláról indulnunk. A P oszlopban szerepel, hogy egy hallgató összesen hány gyakorlaton vett részt, ezt a számot kell csak elosztani a gyakorlatok számával és megkapjuk a jelenléti arányt, amit cellaformázással százalékos alakban tudunk ábrázoltatni.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



7-2. ábra. Tájékozódási lehetőségek a munkalapfüggvényekről

Az R oszlopba adjuk meg, hogy a hallgatók teljesítették-e az ellenőrző mérést. Az ellenőrző mérést akkor tejesítették, ha az arra kapott osztályzatuk, ami az O oszlopban van eltárolva, legalább kettes lett, tehát a HA függvénnyel a következő formulát tudjuk felírni: =HA(O2>=2; "Igen", "Nem")

Az S oszlopba adjuk meg, mi a legjobb gyakorlati jegy, amit a hallgatók kaptak. Most egyszerűen a MAX munkalapfüggvényt kell használnunk a gyakorlati jegyek tartományára, ahogy az a 7-3. ábrán is látszik.

	A	K	L	M	N	O	P	Q	R	S
1	**Név**	**10**	**11**	**12**	**13**	ellenőrz	jelenlét	százalék	EM	legjobb
2	Bárány Róbert	4	5	5	1	3	=DARAB(B2:N2)	=P2/13	=HA(O2>1,"Igen", "nem")	=MAX(B2:N2)
3	Békésy György	3	5	3		5	=DARAB(B3:N3)	=P3/13	=HA(O3>1,"Igen", "nem")	=MAX(B3:N3)
4	Gábor Dénes	5	3	5	2	5	=DARAB(B4:N4)	=P4/13	=HA(O4>1,"Igen", "nem")	=MAX(B4:N4)
5	Harsányi János	3	4	4	3	2	=DARAB(B5:N5)	=P5/13	=HA(O5>1,"Igen", "nem")	=MAX(B5:N5)

7-3. ábra. Az első négy részfeladat megoldása.

Kicsit nehezebb a helyezünk, ha nem a legjobb, hanem a második legjobb osztályzatot szeretnénk megkapni a T oszlopban, hiszen a MAX függvényt nem használhatjuk és nincs olyan munkalapfüggvény,

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

ami egy halmaz vagy tartomány második legnagyobb elemét adná vissza. Viszont szerencsére a NAGY függvény megfelelő paraméterezés mellett használható erre. A "=NAGY(B2:N2, 2)" formula az első hallgató második legjobb jegyét adja vissza.

Menjünk azonban egy fokkal tovább. Mi a hallgatók harmadik, negyedik, ötödik legjobb jegye? Természetesen megtehetnénk, hogy a fenti formulát módosítgatjuk átírva a 2-t először 3-ra az U oszlopban, majd 4-re a V oszlopban, és 5-re a W oszlopban, de ez fölöttébb macerás. Az lenne az igazán jó, ha a formulát egyszer kéne csak beírni és utána a formula másolásával (jobb alsó sarkának elhúzásával) elő tudnánk állítani az összes keresett formulát. Akár a harmadik, vagy a negyedik legnagyobb elemet is keressük, mindig ugyanazon gyakorlati jegyek között keresünk, tehát B2:N2 tartománynak nem szabad megváltoznia, ha jobbra másoljuk a formulát, azaz az oszlopok abszolút pozíciójának változatlannak kell lennie. Ezzel szemben lefele másolva a formulát azt szeretnénk, hogy az adott sorhoz tartozó jegyeket vegye figyelembe a formulánk, tehát B2:N2 tartomány a formulához viszonyított relatív pozíciójának változatlanak kell lennie. Összefoglalva a NAGY függvény első argumentumába a $B2:$N2 tartomány-kifejezést kell írnunk. Mi a helyzet azonban a NAGY függvény második argumentumával, azzal ami meghatározza, hogy hányadik legnagyobb osztályzatra vagyunk kíváncsiak? Több módon is megoldhatjuk, hogy ezt se kelljen a formulánk másolása után módosítani. Egy lehetséges megoldás mutat a 7-4. ábra.

	A	T	U	V	W	X
1	**Név**	2	=T1+1	=U1+1	=V1+1	=W1+1
2	Bárány Róbert	=NAGY($B2:$N2,T$1)	=NAGY($B2:$N2,U$1)	=NAGY($B2:$N2,V$1)	=NAGY($B2:$N2,W$1)	=NAGY($B2:$N2,X$1)
3	Békésy György	=NAGY($B3:$N3,T$1)	=NAGY($B3:$N3,U$1)	=NAGY($B3:$N3,V$1)	=NAGY($B3:$N3,W$1)	=NAGY($B3:$N3,X$1)
4	Gábor Dénes	=NAGY($B4:$N4,T$1)	=NAGY($B4:$N4,U$1)	=NAGY($B4:$N4,V$1)	=NAGY($B4:$N4,W$1)	=NAGY($B4:$N4,X$1)
5	Harsányi János	=NAGY($B5:$N5,T$1)	=NAGY($B5:$N5,U$1)	=NAGY($B5:$N5,V$1)	=NAGY($B5:$N5,W$1)	=NAGY($B5:$N5,X$1)

7-4. ábra. A legjobb jegyek megkeresése egyetlen formula másolásával

Az ábrán a NAGY második argumentumát mindig a fejléc sorból veszi a formula. A dollár jel a sor koordinátája előtt biztosítja, hogy lefelé másolás után is az első sorra mutat a második argumentumban megadott cím, míg a dollár hiánya az oszlop betűjele előtt biztosítja, hogy jobbra másolás után is mindig az adott oszlop fejlécére mutat a második argumentumban megadott cím. A7-4. ábrán látható továbbá, hogy hogyan lehet egyszerűen a fejléceket beállítani úgy, hogy megadják, hogy hányadik legnagyobb osztályzatot tartalmazza az adott oszlop. Egyszerűen az előző fejléc értékhez kell hozzáadni egyet. Ez a módszer is azért kényelmes, mert a cella másolásával gyorsan le tudjuk gyártani a szükséges oszlopfejléceket.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

A fentiek után már könnyedén meg tudjuk válaszolni, hogy mi a hallgatók legjobb kilenc osztályzatának átlaga. Az "=ÁTLAG(S2:AA2)" formulában a korábban kiszámolt részeredményeket használjuk fel.

Az AC oszlopban adjuk meg, hogy van-e a hallgatóknak legalább kilenc gyakorlata, amit legalább kettessel teljesítettek. Ha észrevesszük, hogy akkor van egy hallgatónak kilenc, legalább elégségessel teljesített gyakorlata, ha a kilencedik legjobb gyakorlati jegye legalább kettes, akkor könnyen adódik a korábbi számításainkat felhasználó megoldás. Emlékezzünk, hogy az AA oszlopban tároltuk a hallgatók kilencedik legjobb jegyét, így a keresett formula: =HA(AA2>=2, "van", "nincs")

Ha megnézzük a táblázatunkat (7-5. ábra), azt láthatjuk, hogy néhány sorban nem a "van" vagy a "nincs" szó szerepel az AC oszlopban, hanem a hibát jelző "#SZÁM!" felírat. Miért van ez? Azért mert néhány hallgatónak nincsen kilenc gyakorlata, így nem értelmezhető a kilencedik legjobb osztályzata sem. Mit lehet ilyen esetben tenni? Jelen esetben ezeket a hibákat figyelmen kívül fogjuk hagyni, de a HAHIBA munkalapfüggvénnyel tudjuk ezeket az eseteket kényelmesen lekezelni.

	A	X	Y	Z	AA	AB	AC	AD
1	**Név**	6	7	8	9	legjobb 9 átlaga	van 9 gyakorlata?	
2	Bárány Ró	4	3	3	3	4.222222222	van	
3	Békésy Gy	5	5	4	3	4.666666667	van	
4	Gábor Dén	5	4	4	3	4.555555556	van	
5	Harsányi Já	4	4	4	4	4.444444444	van	
6	Hevesy Gy	4	2	2	#SZÁM!	#SZÁM!	#SZÁM!	
7	Kertész Im	5	4	4	3	4.555555556	van	
8	Lénárd Fül	#SZÁM!	#SZÁM!	#SZÁM!	#SZÁM!	#SZÁM!	#SZÁM!	
9	Oláh Györg	4	4	4	4	4.222222222	van	
10	Polányi Já	3	3	3	1	3.888888889	nincs	
11	Szent-Gyö	#SZÁM!	#SZÁM!	#SZÁM!	#SZÁM!	#SZÁM!	#SZÁM!	
12	Wigner Je	5	5	5	5	5	van	
13	Zsigmondy	5	5	5	3	4.777777778	van	

7-5. ábra. A feladat megoldása néhány részfeladat után

Az AD oszlopban adjuk meg, hogy a hallgatók kaphatnak-e félévvégi jegyet. Az egyszerűség kedvéért most nem szövegesen fogjuk ezt megadni, azaz nem a "kaphat", "nem kaphat" értékeket adjuk a cellák értékének, hanem logikai kifejezésként adjuk meg a válaszunkat. Akkor jár a félévvégi jegy, ha legalább elégségessel teljesített a hallgató kilenc gyakorlatot és átment az ellenőrző mérésen is, tehát a használandó formula: =ÉS(AA2>=2, O2>=2)

Végezetül az AE oszlopban adjuk meg a hallgatók féléves jegyét. Emlékeztetőül a hallgató féléves jegye a kilenc legjobb órai gyakorlatra kapott jegy és az ellenőrző mérésre kapott jegy kerekített átlaga, ahol az ellenőrző mérés jegye 50 százalékos súllyal szerepel. Ezt a végső feladatot is részfeladatokra lehetne

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

bontani, de remélhetőleg enélkül is követhető lesz a megoldás. Természetesen most is felhasználjuk a korábbi részfeladatok megoldását. Hogyan lehet a súlyozott átlagot kiszámolni? A kilenc legjobb gyakorlat átlagát az AB oszlopban már kiszámoltuk, ennek és az ellenőrző mérésnek kell venni az átlagát: =ÁTLAG(AB2, O2). Mindez azonban akkor számít csak, ha a hallgató amúgy teljesítette a tantárgy követelményeit. Ez az AD oszlopban néztük meg. Tehát a végső formula a következő: =HA(AD2, ÁTLAG(AB2, O2), 1)

Utolsó simításként pedig az AE oszlop cellaformázásnál állítsuk be, hogy a számok tizedes jegyeit ne mutassa az Excel, így már tényleg az Indexbe vagy a Neptunba beírható osztályzatokat kapunk.

8. Makrók használata

Makrók rögzítése, visszajátszása, gyorsbillentyűhöz rendelése. Relatív, abszolút címzés a markórögzítés során. Makrók forrásának megtekintése

Az Excellel végzett munka során gyakran elő szokott fordulni, hogy már egy meglévő munkafüzeten kell szisztematikus módosításokat végrehajtani. Tegyük fel, hogy egy 400 sorból álló munkalap minden egyes során ugyanazt a változtatást elvégezni. Ebben az esetben a 400 sor átírása egyrészt időigényes, másrészt könnyű hibázni a monoton munkában. Ekkor jöhet jól, ha a munkafolyamat egyik lépéssorozatát rögzítjük, és egy gyorsbillentyűhöz rendeljük, majd később csak ezt gyorsbillentyűt kell 400 megnyomnunk. (A következő fejezetekben majd azt is elsajátítjuk, hogy hogyan lehet pusztán egyetlen gombnyomással a kellő számú lépésszer visszajátszani a korábban rögzített lépéssorozatot.)

Az Excel nyelvén egy lépéssorozatot makrónak hívjuk és a lépéssorozat felvételét markórögzítésnek és a makró visszajátszását a makró futtatásának. Elképzelhetjük a makrókat úgy is, hogy a rögzítésnél valaki egy videokamerával felveszi, hogy milyen billentyűket milyen sorrendben nyomtunk meg, hogy az egeret merre mozgattuk és hogy mikor klikkeltünk. Majd a makró futtatása során a videót rögzítő személy a felvételek alapján helyettünk a rögzített sorrendnek megfelelően leüti a billentyűket, amiket mi is leütöttünk, arra mozgatja az egeret, amerre mi is mozgattuk, stb. Ez az analógia segíthet megérteni a makrók működési elvét, azonban ennél egy kicsit árnyaltabb a kép.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

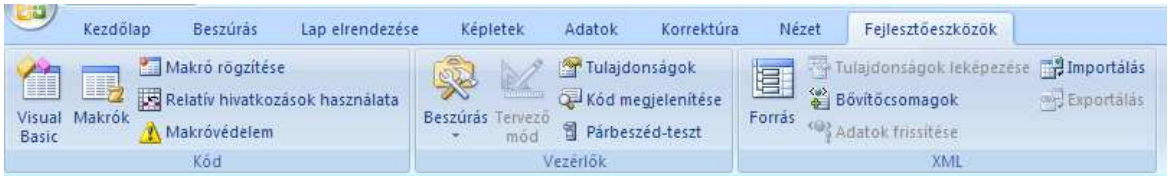
	A	B	C	D	E	F	G	H	I	J	K	
1	Név	Lakcím	dátum	szak	iratkozás éve	tanszék	témavezető neve	utolsó aktív félév	passzív félév kérése	ideiglenes diákig.	Felzárkóztató félév ké	egyéb:
2	Bárány Róbert	Százholdas pagony	2010.02.01	esza	2008. szepter	TMIT	A. A. Milne	2009. I			x	
3	Békésy György	Elm utca	2010.02.02	Info	2009. szepter	TMIT	Freddy Kruger	2009. II.		x		
4	Gábor Dénes	Futrinka utca	2011.01.21	esza	2010. szepter	TMIT	Kuvik Kelemen	2010. II.				soronkívüli
5	Harsányi János	Sesame Street	2011.01.14	esza	2010. szepter	TMIT	Cookie Monster	2010. II.			x	
6	Hevesy György	Pál utca	2011.02.11	esza	2010. szepter	TMIT	Molnár Ferenc	2010. II.		x		
7	Kertész Imre	Champs-Élysées	2010.12.21	esza	2010. szepter	TMIT	Marie Antoinette	2010. II.		x		

8-1. ábra. Hallgatói kérvények. Mintaadat makró-rögzítéshez

Nézzük meg a 8-1. ábrán található munkafüzetet. A táblázat hallgatói kérvények adatait tartalmazza. Egy sor egy kérvénynek felel meg. Az A-H oszlopok a hallgatóra vonatkozó adatokat adnak meg, míg az I, J, K oszlopokban X szerepel, ha a hallgató kérelmének tárgya az oszlopfejlécben található. Az L oszlop egyéb kérelmek szöveges megadására szolgál.

Tegyük fel, hogy ebben a táblázatban tömeges módosításokat kell végrehajtanunk, mert a hallgatók nagy számban szeretnének népszerű, szegedi "Béke Tanszékre" átiratkozni. Tehát feladatunk rögzíteni egy makrót, ami feltételezi, hogy kezdetben az A oszlopban van a kurzor, ekkor az adott sorhoz tartozó tanszék értéket lecseréli "Béke Tanszék"-re, és az igényt passzív félév kérésére módosítja (azaz a J, K, L oszlopok értékét törli és I-be egy X-et tesz); végezetül a következő sor A oszlopában hagyja a kurzort.

Noha nem feltétlenül szükséges, mégis hasznos megint bekapcsolni a 3. fejezet végén leírtaknak megfelelően fejlesztőeszközök szalagot (8-2. ábra).



8-2. ábra. A Fejlesztőeszközök szalag

Tehát helyezzük a kurzort az A2-es cellába, majd a Fejlesztőeszközök szalag Kód csoportjában klikkeljünk a Makró rögzítése gombra. Ekkor egy ablakban megadhatjuk a rögzíteni kívánt makró adatait. Megadhatjuk a makró nevét (például tanszék_váltás). Továbbá definiálhatunk egy, a makróhoz kapcsolódó gyorsbillentyűt, aminek a lenyomása majd lefuttatja a most rögzítendő makrót (ez lehet például control+q). Továbbá megadhatjuk a makró helyét, azaz hogy hova mentse le az Excel a makrónkat. Jelen esetben érdes az alapértelmezett "Ebben a munkafüzetben" opciót megtartani, mert a tanszékváltós makrónknak csak ezeket az adatokon, csak az itt használatos adatstruktúrán van értelme. Végezetül írhatunk egy hosszabb leírást a makró működéséről. Könnyen előfordulhat, hogy egy

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

hosszabb munkamenet során jó néhány makrót rögzítünk, ezért legalább a makrók nevét mindenképpen tanácsos az alapértelmezett Makró1, Makró2, stb. névről olyanra változtatni, ami utal a makró által elvégzett tényleges feladatra. Ha tehát megfelelően kitöltöttük a Makrórögzítés ablak által kért adatokat, akkor az OK megnyomásával végre elkezdhetjük a makrónk rögzítését.

A rögzítés első lépéseként azt kell eldöntenünk, hogy bekapcsoljuk-e a Relatív hivatkozások használatát (8-2. ábra). A formulákban használatos relatív, illetve abszolút címzéshez hasonlatosan ez az opció arra szolgál, hogy a rögzíteni kívánt munkafolyamatot az Excel az aktuális cellához képes rögzítse, vagy az egyes lépések globális pozícióját jegyezze fel. Esetünkben a kurzor az A2-es cellán áll, azaz az A2-es cella az aktív cella. Ha nem kapcsoljuk be a relatív hivatkozásokat és ebben a sorban átírjuk a tanszék nevét Béke tanszékre, és elvégezzük a többi módosítást is, akkor a makró futtatása során (control-q billentyűzetkombináció lenyomása után) megint a második sorban fogja a makró átírni a tanszék nevét. Tehát jelen esetben be kell kapcsolnunk a relatív hivatkozások használatát.

Ott tartunk, hogy az A2-es cella az aktív, és relatív hivatkozásokkal rögzítjük a makrónkat. Vagy kurzormozgató billentyűkkel, vagy az egérrel jelöljük ki az F2-es cellát és írjuk át a tartalmát "Béke tanszékre", haladjunk tovább és az I2-es cellába helyezzünk egy "x" karaktert. Majd töröljük ki a J2, K2, L2 cellák értékét. Álljunk meg egy szóra! Mi értelme kitörölni a J2-es cella tartalmát, amikor az már most is üres? Valóban, a második sorban ennek nincs túl sok értelme, azonban a makrórögzítés során egy olyan munkafolyamatot kell rögzítenünk, ami minden lehetséges adatra, minden lehetséges hallgatói kérvényre helyesen fut majd le. Ha azt szeretnénk, hogy a makró helyesen fusson le az A3-as cellából indítva is, már most ki kell törölnünk az üres J2-es cellát, mert a J3-as cella nem üres.

A rögzíteni kívánt munkafolyamat utolsó lépéseként mozgassuk a kurzort az L2-es cellából át az A3-as cellába, majd ugyanarra a gombra klikkelve, amivel a rögzítést elindítottuk, fejezzük be a makrónk rögzítését. Figyeljük meg, hogy a makró felvételét az A2-es cellából kezdtük és az A3-asban fejeztük be. Relatív hivatkozásokat használtunk, ezért ha most megnyomjuk a makróhoz rendelt gyorsbillentyűt (control-q), vagy a Fejlesztőeszközök szalag Makrók ikonjára klikkelünk és kiválasztjuk a most felvett makrót, akkor lefut a makrónk. A markón visszajátszása átírja a harmadik sorban a tanszék nevét, kitöröli a megfelelő cellák tartalmát és az A4-es cellába helyezi a kurzort. Ez azért praktikus, mert a gyorsbillentyű ismételt lenyomásával könnyedén fel tudjuk dolgozni a munkalap adatait.

Általánosságban is elmondható, hogy érdemes úgy rögzíteni a makrókat, hogy kényelmesek legyenek ismételt futtatások használatakor.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Végezetül a Fejlesztőeszközök szalag Makró ikonján keresztül válaszuk ki a makrónk szerkesztését. Ekkor a Microsoft Visual Basic Editor ablak fog megjelenni benne a makrónkhoz tartozó Basic programkód (8-3. ábra).

```
Sub tanszék_váltás()
'
' tanszék_váltás Macro
'
' Keyboard Shortcut: Ctrl+q
'
    ActiveCell.Offset(0, 5).Range("A1").Select
    ActiveCell.FormulaR1C1 = "Béke tanszék"
    ActiveCell.Offset(0, 3).Range("A1").Select
    ActiveCell.FormulaR1C1 = "x"
    ActiveCell.Offset(0, 1).Range("A1").Select
    Selection.ClearContents
    ActiveCell.Offset(0, 1).Range("A1").Select
    Selection.ClearContents
    ActiveCell.Offset(0, 1).Range("A1").Select
    Selection.ClearContents
    ActiveCell.Offset(1, -11).Range("A1").Select
End Sub
```

8-3. ábra a tanszék_váltás makró forráskódja

Anélkül, hogy túlzottan értenénk bármit is a forráskódból, pusztán némi angol nyelvtudással kisilabizálhatunk egy két dolgot. A "Béke tanszék" és "x" stringeket könnyen észrevehetjük. Mindkét string egy értékadás jobb oldalán szerepel, viszont a bal oldal mindkét esetben azonos. Azt tudjuk, hogy különböző cellák értékét cseréli le a makrónk Béke tanszékre és x-re. Így sejthetjük, hogy az értékadás bal oldala az aktuálisan kijelölt cellát, az aktív cellát (ActiveCell) jelöli. Mi történik a két értékadás között? A makrórögzítés során az F2-es cellából az I2-es cellába váltottunk, tehát a két értékadás közti egyetlen sornak valahogy ezt a cellaváltást kell leírnia. És valóban: az I2-es cella pontosan három cellányira van az F2-es cellától.

Tovább is lehetne folytatni ennek, a most még ismeretlen programnyelven írt rövid makrónak az elemezgetését, de most érjük be annyival, hogy a felvett makrók Basic programnyelven kerülnek rögzítésre. Ezeknek a makróknak a forrását meg tudjuk tekinteni és azokat, ha szeretnénk, módosíthatnánk is.

9. Egyszerű Visual Basic programok

Basic nyelvi elemek: eljárások, While/Until ciklus, If szintaktikája, Not, Mod, értékadások. Programminták az aktív cella mozgatására épülve: ActiveCell, IsEmpty, Cells, Offset, Select, Row, Column

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

A Python programozási nyelvet ismerőknek a Visual Basic programnyelv szintaktikája nem lesz teljesen ismeretlen. Van egy-két hasonlóság és egy pár alapvető különbség is a két programozási nyelv között. Azonban nem célunk a Basic minden részletének megismerése, megelégszünk annyival, amennyivel Excel, Word és egyéb vezérlési feladatainkat meg tudjuk oldani. Ugyanis a Basic egy általános programozási nyelv, de számunkra azért érdekes, mert a különböző Office alkalmazások programozási interfésze könnyedén elérhető vele.

Rögzítsük az előző fejezet során használt makró egyszerűbb változatát, ami csak a tanszéket írja át az F oszlopban. Az makróhoz felvett programkód a 9-1. ábrán látható.

```
Sub Tanszek_atiras()
'
' Tanszek_atiras Macro
' az A oszlopból indítvan átírja a tanszéket
' "Béke tanszék"-re és következő sorra ugrik
'
' Keyboard Shortcut: Ctrl+q
'
    ActiveCell.Offset(0, 5).Range("A1").Select
    ActiveCell.FormulaR1C1 = "Béke tanszék"
    ActiveCell.Offset(1, -5).Range("A1").Select
End Sub
```

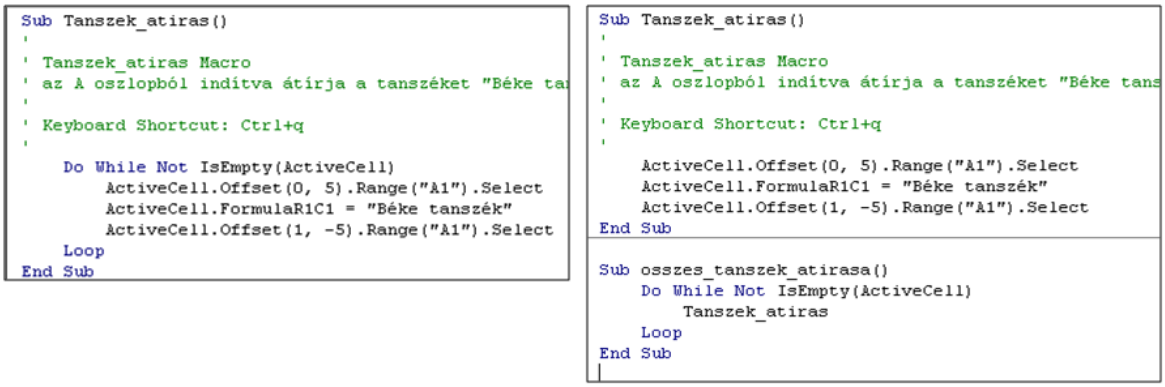
9-1. ábra. A tanszékátíró makró egyszerűbb változata

A kommentek az aposztróf karakterrel vannak bevezetve. Az aposztróftól kezdően a sor végéig a Basic értelmező mindent figyelmen kívül hagy. A kommentek nem a számítógépnek, hanem a programozónak szólnak.

A Python def kulcsszavához hasonlóan a Basicben a Sub kulcsszóval kell bevezetni egy eljárás definiálását. A Sub kulcsszó után az eljárás neve, majd az eljárás argumentumai következnek. Jelen példánkban a Tanszek_atiras eljárásnak nincsenek argumentumai. Az eljárás törzsében szerepelnek az eljárás meghívásakor lefuttatandó utasítások. Szintén a Pythonhoz hasonlóan egy sorban egy utasítás lehet. Végezetül az eljárás végét az End Sub kulcsszó zárja.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



```
Sub Tanszek_atiras()
'
' Tanszek_atiras Macro
' az A oszlopból indítva átírja a tanszéket "Béke ta
'
' Keyboard Shortcut: Ctrl+q
'
    Do While Not IsEmpty(ActiveCell)
        ActiveCell.Offset(0, 5).Range("A1").Select
        ActiveCell.FormulaR1C1 = "Béke tanszék"
        ActiveCell.Offset(1, -5).Range("A1").Select
    Loop
End Sub
```

```
Sub Tanszek_atiras()
'
' Tanszek_atiras Macro
' az A oszlopból indítva átírja a tanszéket "Béke tans
'
' Keyboard Shortcut: Ctrl+q
'
    ActiveCell.Offset(0, 5).Range("A1").Select
    ActiveCell.FormulaR1C1 = "Béke tanszék"
    ActiveCell.Offset(1, -5).Range("A1").Select
End Sub

Sub osszes_tanszek_atirasa()
    Do While Not IsEmpty(ActiveCell)
        Tanszek_atiras
    Loop
End Sub
```

9-2. ábra. A két megoldás a hallgatói kérvényeket tartalmazó táblázat tanszék adatainak átírására

Tegyük fel, hogy a tanszék nevét a hallgatói kérvényeket tartalmazó táblázat összes sorában át szeretnénk írni. Persze ezt a feladatot megoldhatnánk úgy is, hogy az aktív cellát az A1-re állítva sokszor lenyomnánk a control+q gyorsbillentyűt. A 9-2. ábra két olyan megoldást mutat, amikor elég pusztán egyetlen egyszer elfuttatni egy makrót az A2-es cellából indítva és a táblázat összes sora módosul. Az ábra bal oldalán lévő eljárás az általunk rögzített makró egyszerű módosítása: az eljárás törzsét változatlanul hagyva egy ciklus magjává tettük. A While ciklus addig hajtja vége a ciklus magot, ameddig a ciklusutasítás feltétele még igaz marad. Jelen esetben ez a "Not IsEmpty(ActiveCell)" logikai kifejezés vizsgálatát jelenti. Az IsEmtpy függvény igaz értékkel tér vissza, ha az argumentumában szereplő tartomány-kifejezés által meghatározott terület üres. Az ActiveCell pedig az aktív cellára mutató tartomány-kifejezés. A Not pedig a mögötte lévő logikai kifejezés negáltját állítja elő. Tehát a While ciklus magja mindaddig végrehajtódik, ameddig az aktív cella nem lesz üres, azaz ameddig az A oszlopban van valami.

A 9-2. ábra jobb oldalán található programkód szintén átírja az összes tanszék nevét, azonban most a korábban rögzített Tanszek_atiras nevű makrót nem módosítottuk. Tudjuk, hogy ez a makró az A oszlopból indítva átírja az adott sorban a tanszék nevét, és a következő sorra ugrik. Az újonnan írt osses_tanszek_atiras nevű makrónk leprogramozása során nem kell ennél többet tudnunk a Tanszek_atiras eljárásról, nem kell tudnunk, hogy a Tanszek_atiras hogyan írja át az F oszlopban a tanszék nevét és hogyan helyezi az aktív cellát a következő sor A oszlopába. A korábbihoz hasonló While ciklus ciklusmagja most egyszerűen meghívja a Taszek_atiras eljárást.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Ez a második megoldás megint egy jó példa volt egy problémának a kisebb részfeladatokra bontásával történő megoldására. Az összes sor átírását úgy oldottuk meg, hogy készítettünk egy, egy sort átíró makrót; meg egy olyat, ami a táblázat minden sorára meghívja az egy sort átíró makrót.

- **Do While** *kifejezés*
  parancs
  parancs
  Loop
- **Do**
  parancs
  parancs
  Loop While *kifejezés*

- **Do Until** *kifejezés*
  parancs
  parancs
  Loop
- **Do**
  parancs
  parancs
  Loop Until *kifejezés*

9-3. ábra. A négyféle ciklusutasítás szintaktikája

A 9-3. ábra a Basicben használható négy különböző ciklusutasítás szintaktikáját mutatja be. Eddig a bal felső formát használtuk. Itt a Do While ciklus a ciklusmagot mindaddig lefutatja, ameddig csak a ciklus feltétel igaz marad. Ehhez képest a jobb felső, Do Until forma annyiban különbözik, hogy a ciklusmagot addig hajtja végre, ameddig a ciklus feltétele igazzá nem válik. A Do While forma egyszer sem hajtja végre a ciklusmagját, ha a ciklus feltétele az első kiértékeléskor hamis lesz, mert előbb vizsgálja meg a kilépési feltételt és ennek függvényében hajtja végre a ciklusmagot. Ezzel szemben az alsó Loop While és Loop Until ciklusutasítások előbb hajtják végre a cikklusmagot és csak azután vizsgálják meg a ciklus kilépési feltételét. Mikor melyik változatot használjuk? Ez sok mindentől függhet. Általánosságban az lehet mondani, hogy a While/Until variánsok közül azt a ciklusszervezési formát használjuk, ami az adott esetben a természetes gondolatmenetünknek jobban megfelel. Amit az egyikkel el lehet érni, azt a másikkal is, egyszerűen csak negálni kell a kilépési feltételt. A Do While és a Loop While variánsok közötti választáskor pedig a feltételvizsgálat ideje alapján döntsünk.

A ciklusutasításoknál egyszerűbb vezérlési szerkezet a feltételes utasítás. Az If utasítás szintaktikája szintén sokban hasonlít a Python feltételes utasításához. A 9-4. ábra bal oldalán egy kellően általános alakban szerepel a szintaktikája. Ha a feltétel1 logikai kifejezés igaz, akkor az akár több parancsból, így több sorból álló "parancsok1" utasításblokk hajtódik végre. Ha a feltétel1 nem igaz, akkor feltétel2 vizsgálata következik, ha az igaz, akkor a parancsok2 utasításblokk hajtódik végre, ha ez sem igaz, akkor

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

a parancsok3 utasításblokk. Az ElseIf tetszőlegesen sokszor megismétlődhet és az Else ág is elhagyható. Az ábra jobb oldalán látható példa az i változó értékének függvényében állítja be az s változó értékét.

```
If feltétel1 Then
        parancsok1
ElseIf feltétel2 Then
        parancsok2
Else
        parnacsok3
End If
```

```
If i > 1 Then
   s = “sok”
Else
   s = “kevés”
End If
```

9-4. ábra. A feltételes utasítás szintaktikája és egy példa a használatára

Térjünk még egyszer vissza a 9-1. ábrán látható makróra. Itt a feladatunkat úgy oldottuk meg, hogy az aktív cellát mozgattuk és aktív cellának adtunk értéket. Sok Excel automatizálási feladat megoldható ehhez hasonló megközelítéssel. Az ilyen feladatok megoldása során csak arra kell gondolni, hogy mi mit csinálnánk, amikor manuálisan a kurzormozgató-billentyűkkel oldanánk meg a feladatunkat. Az alábbi Excel változók hasznosak lehetnek az ilyen elvet követő programozáskor.

Az ActiveCell egy olyan tartomány-kifejezés, ami az aktuális munkalap aktív cellájára mutat.

A Cells(sor, oszlop) egy olyan tartomány-kifejezés, ami az aktuális munkalap "sor" és "oszlop" egész számok által meghatározott cellájára mutat.

Ha t egy tartomány-kifejezés, akkor t.Offset(sor, oszlop) egy olyan tartomány-kifejezés, ami a t-hez képest "sor" sorral és "oszlop" oszloppal eltolt tartományra mutat.

Ha t egy tartomány-kifejezés, akkor a t.Select egy olyan utasítás, ami kiválasztja a t által meghatározott tartományt. Ha t egyetlen cellára mutat, akkor ez az utasítás az aktív cellát állítja be t-re. Ennek alapján az "ActiveCell.Offset(0, 5).Select" utasítás az aktuális cellát, vagy más néven a kurzort mozgatja öt oszloppal jobbra.

Ha t egy tartomány-kifejezés, akkor a t.Row illetve a t.Column azt adja meg, hogya t tartomány jobb felső sarka a munkalap hányadik sora illetve oszlopa. Tehát például ha a kurzor az A4-es cellában van, akkor az ActiveCell.Row értéke 4 és az ActiveCell.Column értéke 1.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Nézzünk most a fentiekre egy kellően összetett példát. Tegyük fel, hogy van egy munkalapunk, amiben az A oszlop üres, viszont a szomszédos oszlopokban számok találhatók. Ezek után legyen az, a nem kimondottan a való életből vett, megoldandó feladatunk, hogy az A oszlopba kell gyűjtenünk az adott sor legkisebb értékét és a legkisebb számot tartalmazó cella értékét meg kell növelnünk eggyel. A 9-5. ábra bal oldala mutat el lehetséges kiindulási állapotot, a jobb oldali fele pedig erre a kiindulási állapotra a megoldást.

	A	B	C	D
1		6	4	5
2		3	6	33
3		9	11	3

	A	B	C	D
1	4	6	5	5
2	3	4	6	33
3	3	9	11	4

9-5. ábra. Mintaadatok a soronkénti legkisebb szám kiválasztásához és a hozzákapcsolódó megoldás

10. A Visual Basic Editor

A szerkesztő használata, nyomkövetés. Az azonnali ablak, Debug.Print()

A 8. fejezetben már láttuk, hogy kell egy makróhoz tartozó forráskódot megnézni. Az ott leírtakon kívül az Excelből még az Alt-F11 gyorsbillentyű lenyomásával is előhozhatjuk a Visual Basic Editort. Ez az Editor lényegében egy egyszerű szövegszerkesztő, azonban számos olyan funkcióval rendelkezik, ami megkönnyíti a programfejlesztést.

Még a legtapasztaltabb programozókkal által megírt kicsit is nagyobb lélegzetvételű programok vagy csak eljárások sem szoktak elsőre helyesen működik. Ilyenkor nagyon nehéz pusztán a program hibás viselkedésének megfigyeléséből kitalálni a programozási hiba okát. A programfejlesztésnek ezt a jelentős részét kitevő hibakeresési folyamatot szokták debugolásnak2 nevezni.

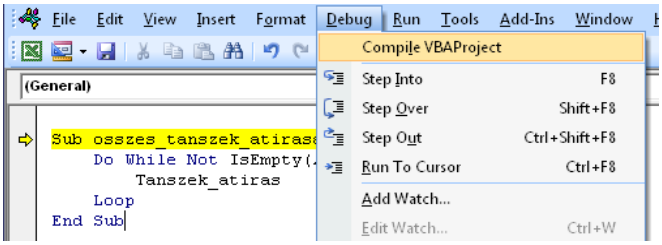
A Fejlesztőeszközök szalag (8-2. ábra) Kód csoportjának Makrók ikonjára klikkelve vagy az Alt-F8 gyorsbillentyű lenyomásának hatására egy ablakban megjelenik az általunk rögzített, vagy írt makrók listája. Itt a kiválasztott makrót nem csak futtatni tudjuk (Indítás menü), hanem a debugolást segítő nyomkövetési módban is tudjuk futtatni a Lépésenként menüponttal. Ekkor a makrót lépésenként hajtja végre az Excel, jobban mondva utasításonként. A soron következő utasítás végrehajtása előtt a Basic

2 A kifejezés eredete még a mechanikus (számító)gépek idejére nyúlik vissza, amikor a hibás működés valóban származhatott a gépbe bekerült bogaraktól.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

értelmező felfüggeszti a program futtását és a Visual Basic Editorban sárga háttérrel kiszínezi az utasítást. A 10-1. ábrán egy eljárásban való belépés látszik.



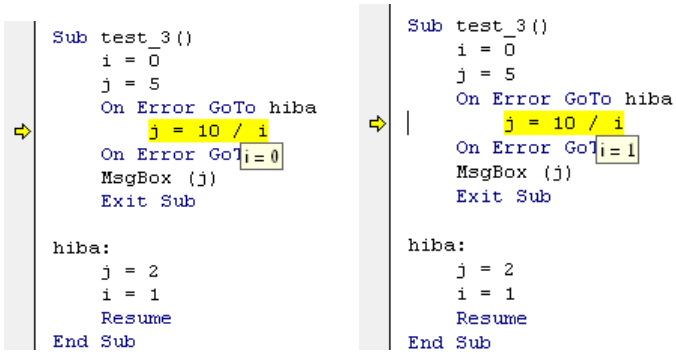
10-1. ábra. Nyomkövetési lehetőségek a Visual Basic Editorban

A felfüggesztett végrehajtást többféleképpen folytathatjuk. A 10-1. ábrán a szerkesztő Debug menüje is látható, amiből kiválasztható a folytatás. Nekünk legtöbbször az F8 gyorsbillentyűvel is elérhető "Step Into"-t érdemes választani. Ennek hatására egy utasítást végrehajt a Basic értelmező, majd megint felfüggeszti a program végrehajtását. Az Excel ablak és a Visual Basic Editor között váltogatva így viszonylag könnyen tudjuk ellenőrizni, hogy az egyes utasításoknak valóban a tervezett hatása van. Ha az általunk elvártól eltérő viselkedést tapasztalunk, akkor le tudtuk szűkíteni a programozási hibát egyetlen egy utasításra, amit remélhetőleg már nem lesz nehéz kijavítani.

A futtatás felfüggesztett állapotában nem csak az Excel ablakban a munkafüzet aktuális állapotát tudjuk megnézni, hanem az Editorban az egérkurzort egy változó fölé húzva egy gyorstipp ablakban közvetlenül a változó mellett leolvashatjuk a változó aktuális értékét. Az ilyen vizsgálatok akkor hasznosak, amikor a programunk éppen vizsgált utasításainak nincs közvetlen látható hatása. A 10-2. ábrán az i változó értékét vizsgálva megállapíthatjuk, hogy a bal oldali esetben a következő, sárgával jelölt utasítás végrehajtása nullával való osztás miatt futási hibát fog eredményezni, míg a jobb oldali esetben ez nem fog bekövetkezi.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



10-2. ábra. Változó értékének vizsgálata nyomkövetés során

Egy nagyobb, több eljárásból álló, nagy adathalmazon futó program végrehajtása során előfordulhat, hogy nagyon sok utasítást kell megvizsgálnunk míg végre megtaláljuk a helytelen eredményt okozó programozási hibát. Ebben az esetben hasznos, ha a "Step Into" helyett másképp folytatjuk a felfüggesztett futtatást. Ha biztosak vagyunk benne, hogy egy eljárás helyesen működik, akkor felesleges annak a végrehajtásának minden egyes lépését külön-külön megvizsgálni. Nem kell belépni az eljárásba, hanem elég átlépni rajta (Step Over), azaz a futtatást elég felfüggeszteni, ha az eljárás végrehajtása befejeződött. Ha biztosak vagyunk benne, hogy a 10-1. ábrán látható Tanszek_atiras eljárásunk helyesen működik, annak végrehajtását átugorhatjuk a Shift-F8 gyorsbillentyűvel.

Szintén hasznos lehet a Control-Shift-F8 billentyűkombinációval elérhető nyomkövetési parancs. Ennek hatására a Basic értelmező befejezi a végrehajtását az aktuálisan nyomon követett eljárásnak, tehát annak az eljárásnak, amelyik a sárga sort tartalmazza.

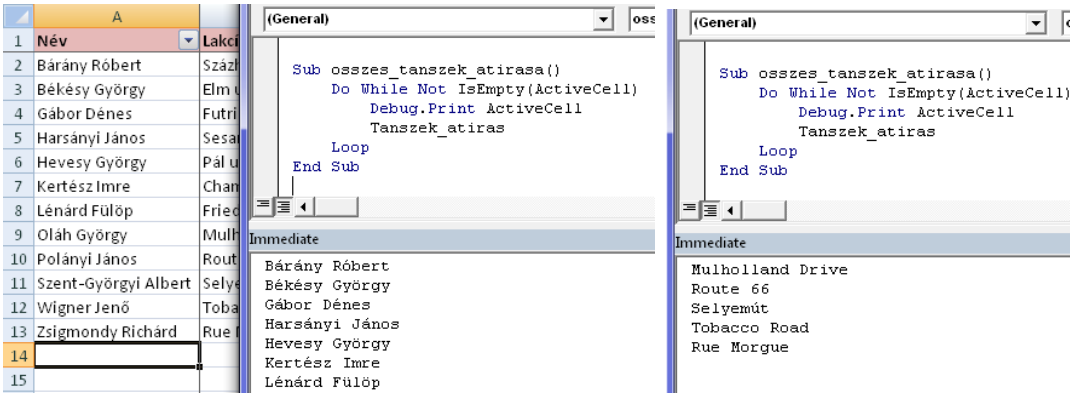
A Control-F8 hatására pedig addig nem fogja megszakítani a programunk futtatását a Basic értelmező, amíg a futtatás a kurzorhoz nem ért. Tehát addig, amíg a következő végrehajtandó utasítás pont a kurzornál található utasítás nem lesz.

Egy nagy és összetett program nyomkövetése során a fenti lehetőségek helyes használatával jól tudunk navigálni a részletesebben megvizsgálni kívánt programsorok között, azonban néha jól lenne az átugrott részek működését is valahogy megvizsgálni. Erre szolgál az úgy nevezett azonnali ablak (Immediate Window) és a debug.Print() utasítás. A debug.Print az argumentumát az azonnali ablakba írja ki, ami a Visual Basic Editor alsó részét található és a normál felhasználók elől az Editorral együtt rejtve van, így csak a programozók láthatják. Használhatjuk naplózási célokra is, de hibakeresésnél is hasznos lehet. A

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

10-3. ábrán az osszes_tanszek_atirasa makrót úgy módosítottuk, hogy egy sor átírása előtt írja ki az azonnali ablakba az aktuális cella értékét. Kétszer futattuk le a makrónkat, az egyik esetben helyesen működött, de a másik esetben hibás végeredményre vezetett. Mi lehetett ennek az oka? Nézzük meg az azonnali ablakot a két futtatás esetén. (Ha nem látszódna az ablak, akkor a Visual Basic Editorból a Control-G gyorsbillentyűvel előhívató.) A jobb oldali esetben látszik, hogy nem a hallgatók nevei szerepelnek az azonnali ablakban, tehát az aktív cella nem az A oszlopban volt, és ezért rossz oszlopokat módosított a makrónk.



10-3. ábra. Példa az azonnali ablak használatára

Az azonnali ablak elnevezése onnan származik, hogy programírás során vagy felfüggesztett futtatáskor rögtön végrehajtható parancsokat adhatunk a Basic értelmezőnek. A kérdőjellel kezdődő sorok esetén az enter leütése után a kérdőjel utáni változó értéke vagy kifejezés eredményét írja a Basic értelmező az adott sort követő sorba. A 10-4. ábra első példájában az aktív cella értékét kérdeztük le. A következő példában egy kifejezés eredményére voltunk kíváncsiak (1+1). Ezek után egy parancsot adtuk ki, ami az aktív cellát mozgatta eggyel lefele. A parancs végrehajtása sikeres volt, hiszen az utolsó példában ismételten lekérdeztük az aktív cella értékét, és az erre adott válasz most az A2-es cella értéke helyett az A3-as cella értéke lett.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.


```
Immediate
 ?ActiveCell
Bárány Róbert
? 1 + 1
 2
ActiveCell.Offset(1, 0).Select
?ActiveCell
Békésy György
```

10-4. ábra. Azonnal végrehajtott parancsok az azonnali ablakban

11. A Basic további nyelvi elemei

Sorfolytató karakter. For ciklus. Visual Basic használata Excel automatizálására: a tartomány (Range) osztály címzési lehetőségei. A Value és a Formula attribútumok közti különbség. Eredmények közlése: Cellában, MsgBox(), Debug.Print().

A Basic nyelvben egy sor egy utasításnak felel meg. Néha azonban összetett kifejezések leírásakor a jobb olvashatóság miatt logikusabb lenne egy parancsot több sorba írni. Erre szolgál az úgynevezett sorfolytató karakter. Ha a sor a sorfolytató karakter zárja, akkor a parancs a következő sorban folytatódik. A Basic programnyelvben a sorfolytató karakter egy szóköz és egy aláhúzás jel. A 11-1. ábrán arra látunk példát, amikor az aktuális cella értékét a tőle balra és a felette lévő cella összegére állítjuk be úgy, hogy magát a parancsot két sorban írjuk le.

```
ActiveCell = ActiveCell.Offset(0, -1) _
           + ActiveCell.Offset(-1, 0)
```

11-1. ábra. Példa a sorfolytató karakter használatára

A While és Until ciklusutasításokkal már a 9. fejezetben találkozhattunk. Ha már előre tudjuk, hogy hányszor szeretnénk a ciklusmagot lefuttatni, a For ciklusutasítást érdemes használnunk. A For parancs általunk használt, egyszerűsített szintaktikája a 11-2. ábrán látható. A Basic értelmező a parancs végrehajtásának első lépésében az egész számokat tartalmazó ciklusváltozót a kezdőértékre állítja, majd egyszer lefuttatja a ciklusmagot. Eztán az értelmező eggyel megnöveli a ciklusváltozó értékét. Ha a ciklusváltozó nem érte el a végértéket, akkor újra lefuttatja a ciklusmagot. Majd ismétli ezt a folyamatot mindaddig, amíg a ciklusváltozó végül eléri a végértéket.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

For *ciklusváltozó = kezdőérték* **To** *végérték*
parancs
Parancs
Next

```
For i = 1 To 3
  For j = 1 To 3
    Debug.Print i, j
  Next
Next
```

11-2. ábra. A For utasítás szintaktikája és egy példa a használatára

A 11-2. ábra jobb oldalán látható példában az i ciklusváltozót használó for ciklus ciklusmagja három sorból áll, ami egy j ciklusváltozót használó for ciklus. A belső, j változós ciklus ciklusmagja egyetlen utasítást tartalmaz, ami az azonnali ablakba írja ki az i és a j változó értékét. A külső ciklus háromszor fog lefutni és rendre 1, 2, és 3 lesz az i ciklusváltozó értéke. A belső ciklus minden egyes i értékre háromszor fog lefutni, tehát összesen kilencszer. Az azonnali ablakban a következő sorok fognak a futtatás után látszódni: "1 1", "1 2", "1 3", "2 1", "2 2", "2 3", "3 1", "3 2", "3 3". A működés nem lenne teljesen világos, érdemes lépésenkénti nyomkövetéssel végignézni a teljes ciklus végrehajtását.

Egy Excel munkafüzetet celláit be tudjuk járni az aktív cella mozgatásával, de for ciklusok és a Cells változó használatával is, hiszen a Cells argumentumába sor és oszlopkoordinátákat kell beírni. Előfordulhat azonban, hogy kényelmesebb a munkalap celláira annak koordinátái helyett a hagyományos Excel használat során megismert címzést használni. A Cells-hez hasonló a Range változó (objektum), aminek egészek helyett egyetlen stringet kell megadni argumentumként. Ez a string bármi lehet, ami a 7-2. ábra felső részének bal oldalán is látható név mezőbe írható. A Range("A1:C5") az A1 és a C5-ös cellák által meghatározott téglalapra mutatató tartomány-kifejezés, a Range("A:A") a teljes A oszlopra mutató tartomány-kifejezés, a Range("3-3") a teljes harmadik sorra, míg a Range("A1:C5, D6:F10") pedig két téglalap uniójára mutató tartomány-kifejezés.

A korábban megismert tartomány-kifejezéseket használó parancsokon kívül az Excel programozási interfésze még számos más parancsot definiál. Az alábbi parancsok gyakran bizonyulnak hasznosnak.

Ha t és c is egy tartomány-kifejezés, akkor t.Copy a t tartományát a vágólapra helyezi. A "t.Copy c" parancs pedig a t tartományát c-re másolja.

Ha t egy tartomány-kifejezés, akkor t.EntireRow egy olyan tartomány-kifejezés, ami a t-t tartalmazó teljes sorra mutat. Hasonlóan a t.EntireColumn a teljes sorra mutat.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

Noha ennek a könyvnek nem célja az Objektum-orientált programozás ismertetése, nem halogathatjuk tovább hogy ejtsünk pár szót a Basic a pontot használó különös szintaktikájáról. Többnyire elegendő annyit tudnunk, hogy a "változónév.parancs" szintaktika azt jelenti, hogy a parancsot a változón illetve a változóban tárolt objektumon kell végrehajtani. Ez a jelölésrendszer azonban lehetőséget biztosít arra, hogy a változótól is függjön, hogy milyen parancs hajtódik végre. Például a t.Copy egy tartományt helyez a vágólapra, ha t egy tartomány-kifejezés; azonban ha t egy munkalap objektum, akkor a t.Copy a munkafüzetbe illeszti a t másolatát.

További előnye a pontot használó szintaktikának, hogy ha az objektumon végrehajtott parancs eredménye egy objektum, akkor azon újra használhatjuk a pont operátort. Erre volt jó példa a ActiveCell.Offset(1, 0).Select parancs: az ActiveCell az aktív cellára mutató objektum, az ActiveCell.Offset(1, 0) az akív cellától jobbra lévő cellára mutató objektum, amin végrehajtott Select utasítás azt a cellát teszi aktívvá.

Míg "rendes" változóknál az egyenlőségjel operátort használhatjuk az értékadásra ("i = 4"), addig, ha azt szeretnénk, hogy egy váltózó egy objektumra mutasson, akkor a "Set változónév = érték" szintaktikát kell használnunk. Például: Set t = c.EntireRow

```
Range("F3").Value = Range("D3").Value*15
Range("F3").Formula = "=D3*15"
```

11-3. ábra. Példa a Value és a Formula közti különbségre

Nézzük meg a 11-3. ábrán látható két parancsot. Ha a D3-as cellában 2 volt, akkor mindkét parancs után az F3-as cellából a 30-as számot olvashatjuk ki. Van azonban egy fontos különbség. Az első esetben az F3-as cellába a 30-as számot írjuk bele, míg a másik esetben az F3-as cellába egy képletet írjunk, aminek az eredménye jelenleg 30. Ha azonban a parancsok végrehajtása után átírjuk a D3-as cellát 1-re, akkor az első esetben az F3-as cella továbbra is 30-at fog tartalmazni, míg a második esetben az F3-as cella továbbra is ugyanazt a képletet fogja tartalmazni, de most a formula 15-re értékelődik ki és nem harmincra. Úgy is fogalmazhatunk, hogy a második esetben egy sorosabb csatolást hoztunk létre a cellák között.

Nézzünk most egy összetett programozási feladatot. Tegyük fel, hogy van egy sok sorból álló munkalapunk, ami belga sörgyárakról és a gyárak által palackozott sörökről tartalmaz adatokat. Ezeknek az adatoknak a segítségével szeretnénk különböző kérdéseket megválaszolni. Például kíváncsiak

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

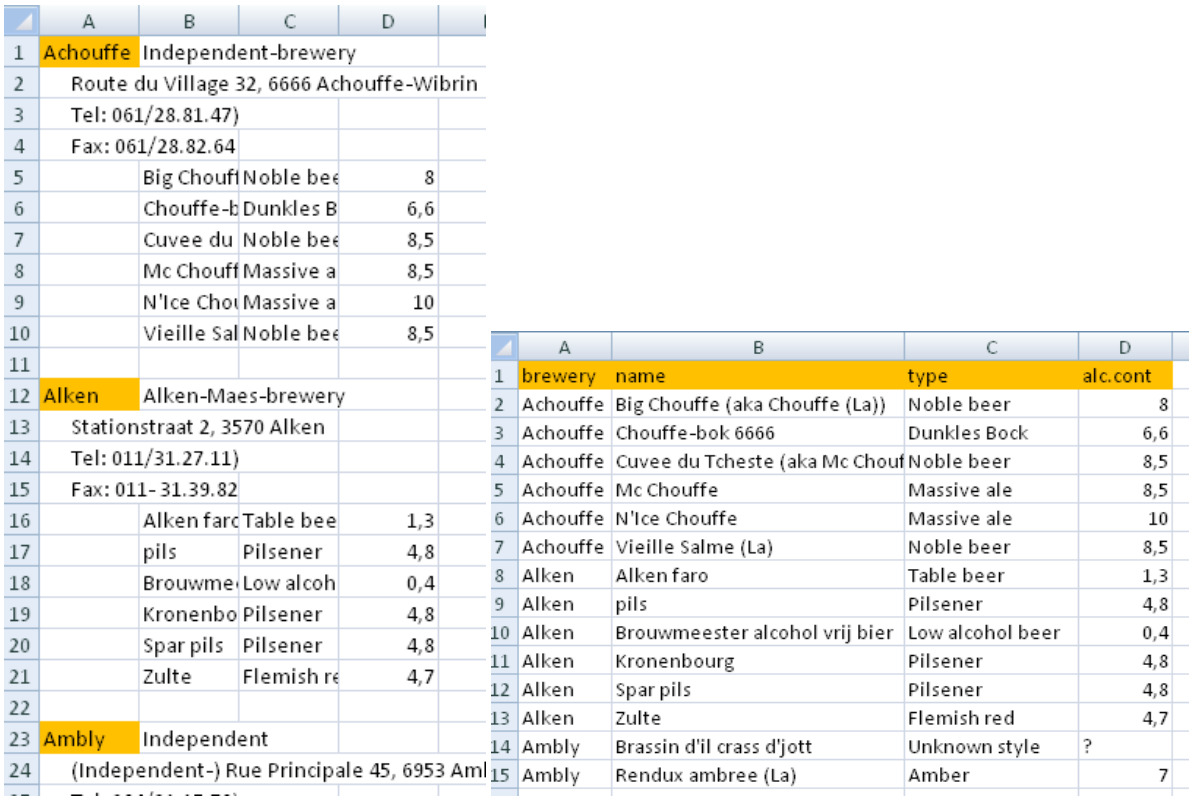
vagyunk arra, hogy az egyes sörtípusukból hány különbözőt lehet kapni, vagy hogy hány százalékos az egyes főzdék által forgalmazott legnagyobb alkoholtartalmú ital. Ezekre a kérdésekre nem könnyű válaszolni, ha a forrásadatok a 11-4. ábra bal oldalán található formátumba vannak megadva. Ha azonban sikerülne egy makrót írnunk, ami a bal oldali adatokat a jobb oldali formátumra hozná, akkor abból kiindulva kimutatásokat tudnánk készíteni (6. fejezet) a kérdéseink gyors megválaszolására.

Az adatok más formátumra való átalakítását többféleképpen is meg lehet oldani. Elsőnek egy olyan megoldást mutatunk be, ami az aktív cella mozgatását és a feladat kisebb részegységekre való bontását használja.

Első lépésként tegyük fel, hogy van már egy olyan masol_egy_gyarat nevű parancsunk, amivel egy teljes gyár adatit át lehet átmásolni a "beers" munkalapról a "3" nevű munkalapra. Azaz ha egy gyárnéven van az aktív cella, például az A1-en vagy az A12-n, akkor a sörgyár adatait átalakítva átmásolja a másik munkalapra és az aktív cellát az A11-es illetve az A22-es cellán hagyja. Ha lenne egy ilyen parancsunk, akkor a teljes feladatot meg lehetne oldani a 11-5. ábrán látható AdatAtlakito nevu makróval. Az első négy utasítással beállítjuk az aktív cellákat a megfelelő kezdőértékre, majd egy Until ciklus segítségével átmásoljuk az összes gyár adatát. A ciklusmagban az aktuális gyár átmásolása után eggyel lejjebb mozgatjuk az aktív cellát, ha ez a cella üres, akkor az azt jelenti, hogy két üres sor van a gyár adatai után, ami a forrásmunkalapon az adatok végét jelzi.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



	A	B	C	D
1	Achouffe	Independent-brewery		
2	Route du Village 32, 6666 Achouffe-Wibrin			
3	Tel: 061/28.81.47)			
4	Fax: 061/28.82.64			
5		Big Chouff	Noble bee	8
6		Chouffe-b	Dunkles B	6,6
7		Cuvee du	Noble bee	8,5
8		Mc Chouff	Massive a	8,5
9		N'Ice Cho	Massive a	10
10		Vieille Sal	Noble bee	8,5
11				
12	Alken	Alken-Maes-brewery		
13	Stationstraat 2, 3570 Alken			
14	Tel: 011/31.27.11)			
15	Fax: 011- 31.39.82			
16		Alken farc	Table bee	1,3
17		pils	Pilsener	4,8
18		Brouwme	Low alcoh	0,4
19		Kronenbo	Pilsener	4,8
20		Spar pils	Pilsener	4,8
21		Zulte	Flemish re	4,7
22				
23	Ambly	Independent		
24	(Independent-) Rue Principale 45, 6953 Aml			

	A	B	C	D
1	brewery	name	type	alc.cont
2	Achouffe	Big Chouffe (aka Chouffe (La))	Noble beer	8
3	Achouffe	Chouffe-bok 6666	Dunkles Bock	6,6
4	Achouffe	Cuvee du Tcheste (aka Mc Chouf	Noble beer	8,5
5	Achouffe	Mc Chouffe	Massive ale	8,5
6	Achouffe	N'Ice Chouffe	Massive ale	10
7	Achouffe	Vieille Salme (La)	Noble beer	8,5
8	Alken	Alken faro	Table beer	1,3
9	Alken	pils	Pilsener	4,8
10	Alken	Brouwmeester alcohol vrij bier	Low alcohol beer	0,4
11	Alken	Kronenbourg	Pilsener	4,8
12	Alken	Spar pils	Pilsener	4,8
13	Alken	Zulte	Flemish red	4,7
14	Ambly	Brassin d'il crass d'jott	Unknown style	?
15	Ambly	Rendux ambree (La)	Amber	7

11-4. ábra. Sörgyárak adatai

Egy gyár adatainak teljes átmásolását három részre bonthatjuk. A masolj_egy_gyarat tehát először átmásolja a gyár nevét (az aktív cella változatlanul hagyásával). Ezután a címzési részre ugrik, majd az aktív cellát a címrész utánra mozgatja (ugord_at_a_cimzest), mivel a címinformációk teljesen elhagyhatók a táblázatos formából (11-4. ábra). Végezetül a gyár összes sörét átmásolja a célmunkalapra.

Mivel a címzési részben csak az A oszlopban vannak adatok és az utána következő részben az A oszlopban nincsenek adatok, ezért az ugord_at_a_cimzest eljárásnak mindaddig lefele kell mozgatnia az aktív cellát, amíg az A oszlopban üres cellába nem ütközik.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

```
Sub AdatAtalakito()
    Sheets("3").Select
    Cells(2, 1).Select

    Sheets("beers").Select
    Cells(1, 1).Select

    Do Until IsEmpty(ActiveCell)
        masolj_egy_gyarat
        ActiveCell.Offset(1, 0).Select
    Loop
End Sub

Sub masolj_egy_gyarat()
    masold_at_a_gyarnevet
    ActiveCell.Offset(1, 0).Select
    ugord_at_a_cimzest
    masold_at_a_soroket
End Sub

Sub ugord_at_a_cimzest()
    Do Until IsEmpty(ActiveCell)
        ActiveCell.Offset(1, 0).Select
    Loop
End Sub
```

```
Sub masold_at_a_soroket()
    Do Until IsEmpty(ActiveCell.Offset(0, 1))
        masolj_at_egy_sort
        ActiveCell.Offset(1, 0).Select
    Loop
End Sub

'átmásolja a gyár nevét.
'az aktív cella a hívás után a
'kiindulási helyén marad
Sub masold_at_a_gyarnevet()
    ActiveCell.Copy
    Sheets("3").Select
    ActiveCell.PasteSpecial
    Sheets("beers").Select
End Sub

Sub masold_at_a_gyarnevet_ha_kell()
    Sheets("3").Select

    If IsEmpty(ActiveCell) Then
        ActiveCell.Offset(-1, 0).Select
        ActiveCell.Copy
        ActiveCell.Offset(1, 0).Select
        ActiveCell.PasteSpecial
    End If

    Sheets("beers").Select
End Sub
```

```
Sub masolj_at_egy_sort()
    masold_at_a_gyarnevet_ha_kell

    'a sör nevének átmásolása
    ActiveCell.Offset(0, 1).Select
    ActiveCell.Copy
    Sheets("3").Select
    ActiveCell.Offset(0, 1).Select
    ActiveCell.PasteSpecial
    Sheets("beers").Select

    'a sör tipusának átmásolása
    ActiveCell.Offset(0, 1).Select
    ActiveCell.Copy
    Sheets("3").Select
    ActiveCell.Offset(0, 1).Select
    ActiveCell.PasteSpecial
    Sheets("beers").Select

    'az alkoholtartalom másolása
    ActiveCell.Offset(0, 1).Select
    ActiveCell.Copy
    Sheets("3").Select
    ActiveCell.Offset(0, 1).Select
    ActiveCell.PasteSpecial
    ActiveCell.Offset(1, -3).Select

    'ugrás a vissza a kiindulási helyre
    Sheets("beers").Select
    ActiveCell.Offset(0, -3).Select
End Sub
```

11-5. ábra. Az adat-átalakítós feladat megoldása részproblémákra való bontással

A masold_at_a_gyarnevet eljárás programozás nélkül is elkészíthető: egyszerűen rögzítenünk kell egy makrót, ami az aktív cella értékét a vágólapra helyezi, átvált a "3" nevű munkalapra beilleszti a vágólap tartalmát, majd visszavált a "beers" munkalapra.

A masold_at_a_soroket az ugord_at_a_cimzest eljáráshoz hasonlóan működik: addig kell átmásolnia az aktív sorban található söradatokat és az aktív cellát eggyel lejjebb mozgatnia, amíg a B oszlopban (azaz aktív cellához képest eggyel jobbra) üres cellába nem ütközik, mert az a sörgyár adatainak a végét jelenti.

Egyetlen sör adatinak az átmásolás (masolj_at_egy_sort eljárás) is nagyjából megoldható a gyárnévnél bemutatott makrórögzítéses módszerrel. Az egyetlen nehézség a gyárnév átmásolása, mivel az nem található meg a "beers" munkalap aktív cellájának közelében. De ennek a részfeladatnak a megoldását a masold_at_a_gyarnevet_ha_kell nevű eljárásra hárítjuk át.

Ha egy sörgyár elsőnek felsorolt sörének adatait másoljuk át, akkor a gyárnévvel már nem kell vesződnie a masold_at_a_gyarnevet_ha_kell nevű eljárásnak, mert azt a masold_at_a_gyarnevet eljárás már átmásolta a "3" munkalap A oszlopába. Ha azonban a sörgyár tovább söreinek az adatait másoljuk át,

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

akkor az A oszlop még üres, ezért a "3" munkalapon az aktív cella feletti oszlopból át tudjuk másolni az aktuális sörgyár nevét.

A fenti programkód némiképp hosszúra sikeredett, de cserébe a programozás során mindig csak kezelhető méretű részfeladatokat kellett megoldani. A megoldás során remélhetőleg sikerült átlátni, részletesebben megérteni a forrás munkalap adatformátumát is, ezért valószínűleg könnyen érthető lesz a 11-6. ábrán látható, ugyanerre a feladatra adott tömörebb megoldás is.

```
Sub AdatAtalakito_2()
    Dim r As Range
    Dim t As Range
    Dim brewery As String

    Set t = Worksheets("3").Range("a1")
    Set r = Range("A1")

    Do Until IsEmpty(r)
        brewery = r.Value
        Set r = r.Offset(1, 0)
        Do Until IsEmpty(r)
            Set r = r.Offset(1, 0)
        Loop
        Do Until IsEmpty(r.Offset(0, 1))
            r.EntireRow.Copy t
            t.Value = brewery
            Set t = t.Offset(1, 0)
            Set r = r.Offset(1, 0)
        Loop
        Set r = r.Offset(1, 0)
    Loop
End Sub
```

11-6. ábra. Második megoldás az adatátalakítós feladatra

A második megoldás nem az aktív cellát mozgatásával dolgozza fel a forrásadatokat, hanem az r változót használja fel arra, hogy eltárolja, hogy éppen hol jár az adatfeldolgozásban. A t változó pedig azt tárolja el, hogy a "3" nevű munkalapon hová kell a következő adatot bemásolni. Az aktuálisan feldolgozás alatt álló sörgyár nevét pedig a brewery változóba tároljuk el. Az első három sor ezen változók deklarálására szolgál, mivel ha egy változóban objektumreferenciát szeretnénk tárolni, szükséges a változót ezt megelőzően deklarálni, ami egyébként a "Dim valtozónév" formával elegendő megtenni.

A következő két sor beállítja a t és az r változók kezdeti értékét. Látható, hogy a külső Until ciklus ciklusmagjának egy futtatása egy sorgyár feldolgozását végzi el. A ciklusmag két Until ciklust tartalmaz. Az első belső ciklus a címrész átugrását végzi el. A második belső ciklus pedig egy sör adatait másolja át

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

a "3" munkalapra méghozzá úgy, hogy egy lépésben átmásolja a sör adatait tartalmazó teljes sort, majd az első oszlopba beírja a brewery változóban eltárolt sörgyár nevét.

Ebben a feladatban a makrónk által elvégzett számítások eredményét egy munkalapon jelenítettük meg. Általában is járható út, hogy egy összetettebb műveletsorozat eredményét a programunk felhasználójával úgy közöljük, hogy az eredményt egy cellába beírjuk. Erre használható például az "ActiveCell = eredmeny" parancs, ami az eredmeny változó értékét az aktív cellába írja. Egy másik megközelítés az, amikor az eredményt egy felugró alakban közöljük a felhasználóval. Például a "MsgBox eredmeny" paranccsal. Végezetül, ha az eredményt nem a felhasználóval szeretnénk megosztani, hanem mi magunk vagyunk rá kíváncsiak, akkor a korábban látott módon az azonnali ablakba is kiírhatjuk. Erre példa a "debug.Print(erendmeny)" parancs. Mindhárom út járható; az a fontos, hogy a programunk felhasználásának figyelembevételével mérlegeljünk a lehetőségek között.

12. Hatékony makrók írása

Felesleges cellamozgatások elkerülése. Frissítés kikapcsolása és a felhasználó tájékozatatása: Application.ScreenUpdating, Application.StatusBar.

A 11-5. ábrán látható AdatAtalakito makrót egy kétezer sorból álló munkalapra lefuttatva az fogjuk tapasztalni, hogy bizony bele telik pár percbe, amíg a makrónk végez a munkájával. Persze ez töredéke annak az időnek, ami ahhoz kell, hogy mi magunk kézzel átírjuk a megfelelő formátumra a forrásadatokat. Akár megelégedéssel is tölthetne el minket az Excel ablakok villódzása és az aktív cella gyors változása, azonban legyünk egy kicsit telhetetlenebbek. Ha például olyan feladatokat szeretnénk automatizálni, amire kézi feldolgozással reményünk sincs, hogy egy emberöltő alatt végezni tudjuk, akkor bizony nagyon fontos, hogy a programunk gyors legyen.

Sok olyan módszert használhatunk, amivel lassú programunk futását némiképp felgyorsíthatjuk. Sőt, általában ezt a fajta megközelítést érdemes használni: írjuk meg a Basic programunkat, makrónkat egy adott feladat megoldására, aztán ha úgy alakul, hogy a megoldásunk túlságosan lassúra sikeredett, akkor próbáljuk úgy csiszolni, hogy megfelelő sebességnövekedést tudjuk elérni. Általában könnyebb megírni egy lassú, de jól működő programot és azt könnyebb felgyorsítani, mint rögtön egy gyors és helyesen működő programot írni.

Ha programunk egy részét makrórögzítéssel írtuk meg, akkor törekedjük arra, hogy elkerüljük a felesleges cellamozgatásokat. Például kezdőknél gyakran előfordulhat, hogy a makrórögzítés során fel

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

kell görgetniük a munkalap tetején található címsorhoz, hogy megnézzék, melyik oszlopban milyen adatok találhatók, majd vissza kell görgetniük a kiindulási állapothoz, hogy folytatni tudják a makrórögzítést. Ha egy ilyen makrót sokszor meghívunk egy másik eljárásból, akkor minden eljáráshívásnál egy felesleges fel- és legördítés hajtódik végre. Ilyenkor egyrészt a rögzítés előtt megpróbálhatjuk végiggondolni, hogy a rögzíteni kívánt munkafolyamatot hogyan lehet minél kevesebb lépésből végrehajtani. Másrészt a felvett makró forráskódjának szerkesztésével kitörölhetjük a felesleges lépéseket. Ez utóbbi azért is hasznos lehet, mert megérthetjük, hogy az egyes lépések milyen Basic parancsokkal valósíthatók meg.

Az Excelbe épített Basic értelmező minden parancs végrehajtása után frissíti az Excel ablakok állapotát. Emiatt láthatjuk a makrónk futása közben, hogy az aktív cella folyamatos változik, hogy az üres cellák a programunk számítási eredményeivel telnek meg. A 12-1. ábra arra mutat példát, hogy hogyan érdemes a parancsonkénti frissítést kikapcsolni. Az Application.ScreenUpdating változó beállításával lehet szabályozni a képernyőfrissítést. Ha biztosak vagyunk benne, hogy egy eljárásunk vagy a teljes programunk helyesen működik, akkor jelentős sebességnövekedést tudunk elérni a részlegesen felfüggeszthetett képernyőfrissítéssel.

```
Sub lassu_makro()
    Application.ScreenUpdating = False

    sok_lassu_parancs

    Application.ScreenUpdating = True
End Sub
```

12-1. ábra. Makró gyorsítása a képernyőfrissítés kikapcsolásával

Ha azonban kikapcsoltuk a képernyőfrissítést és a programunk még így is sokáig fut, akkor a programunk felhasználója nem fogja tudni, hogy mi történik a háttérben. Vajon mennyit kell még várnia? Érdemes-e egyáltalán várnia, vagy a programunk végtelen ciklusba került és soka nem fogja befejezni a működését? A képernyőfrissítés időnkénti ideiglenes visszakapcsolásával tudnánk tájékoztatni a felhasználót a programunk futási állapotáról, de egy teljes képfrissítés túlságosan időigényes lehet. Ilyen esetekben a felhasználó gyors tájékoztatására használható az Excel státuszsora. Az Application.StatusBar változónak értéket adva tudunk a státusz sorba írni. Az "Application.StatusBar = False" paranccsal pedig lemondhatunk a státuszsor használatáról, a parancs kiadása után újra az Excel által szokásosan kiírt státuszüzenetek fognak ott megjelenni. A változó használatára mutatat egy jól testre szabható sémát a

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

12-2. ábra. A sokaig_tart eljárást kell meghívni az aktuális munkalap első száz sorárára. Minden ötödig sor feldolgozása után a státuszsorba írjuk a feldolgozott sor sorszámát.

```
Sub lassu_makro()
    Application.ScreenUpdating = False

    Range("A1").Select
    For i = 1 To 100
        sokaig_tart
        ActiveCell.Offset(1, 0).Select
        If i Mod 5 = 0 Then
            Application.StatusBar = i
        End If
    Next

    Application.StatusBar = False
    Application.ScreenUpdating = True
End Sub
```

12-2. ábra. Példa a státuszsor használatára.

További gyorsítási lehetőség adódhat a megfelelő számítási beállítás kiválasztásából. Erről részletesebben a 15. fejezetben lesz szó.

13. Automatikus dokumentumgenerálás

Dokumentum automatikus előállítása Excel adatokból címtár segítségével. XML-szerű adatokból formázott dokumentum előállítása makró használatával.

Az 5. fejezetben már láthattuk, hogy hogyan lehet egy Excel munkalap és egy dokumentumsablon felhasználásával több dokumentumot előállítani. Most arra látunk példát, hogy egy Excel munkalapból egy dokumentum ismétlődő részeit legenerálni.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

13-1. ábra. Címtárként generált dokumentum

	A	B	C	D	E	F	G	K
1	kat.	típus	Borászat	fajta	évjárat	szárm	űrtart.	ár
2	száraz	fehérbor	Laposa Pir	Badacsonyi Rizl	2008	Badacsony	0,75	146(
3	száraz	fehérbor	Bolyki	Egri királyleán	2009	Eger	0,75	150(
4	száraz	fehérbor	Gál Tibor	S.D. Gloria	2007	Eger	0,75	345(
5	száraz	fehérbor	Németh P	Juhfark	2007	Badacsony	0,75	250(
6	száraz	fehérbor	Figula	Nyerges (szürk	2006	Balatonszőlős	0,5	264(
7	száraz	rosébor	Sike Tamá	Pinot Noir Ros	2009	Eger	0,75	113(
8	száraz	rosébor	Légli	pinot noir, kék	2009	Balatonboglár	0,75	125(
9	száraz	vörösbor	Sike	cabernet sauvi	2005	Eger	0,75	110(

13-2. ábra. Forrásadatok a címtáras dokumentumhoz

A körlevélvarázsló első lépésében a dokumentum típusának általában levelet szoktunk megadni, de néha hasznos lehet eltérő típusok használat. A címtár típussal el tudjuk érni, hogy a forrásadatok egyetlen sablonba kerüljenek bele. Ez természetesen hasznos akkor, ha valóban egy címtárat szeretnénk előállítani az Excelben tárolt személyeket tartalmazó táblázatból, azonban máskor is jól jöhet a címtár típus használata. Tegyük fel, hogy az 13-2 ábrán látható formátumban tárolja egy borbaráti közösség a raktáron lévő borai adatai. a címtár előállítására használható funkció segítségével az 13-1. ábrán látható itallapot könnyedén előállíthatjuk. Sőt, a társaság következő összejövetelére elég csak az Excel fájlt frissíteni és az új itallap rögtön előállítható belőle.

```
<temakiiras><cim>Erő érzékeltetése (force feedback) kognitív infokommunikációs csatornák
alkalmazásával</cim><kod>198</kod><oktatok>Baranyi Péter</oktatok><munkahely>BME
TMIT</munkahely><letszam>3</letszam><kivonat>A feladat célja olyan infokommunikációs algoritmusok
fejlesztése, amelyek hatékony és költségkímélő mester-szolga telemanipulációt tesznek
lehetővé.</kivonat><reszletes>A jelentkező feladata, hogy a nálunk kifejlesztett VirCA (Virtual
Collaboration Arena) nevű 3D virtuális kognitív infokommunikációs platformhoz olyan szoftver
komponenseket fejlesszen ki, amelyek lehetővé teszik a virtuális valóságban végzett telemanipuláció
során fellépő, a valós manipulációban tapasztalható erők visszajelzését. A fejlesztésben lehetőség
van japán, norvég és indiai hallgatókkal közösen dolgozni csoportmunkában.</reszletes></temakiiras>¶

<temakiiras><cim>Kognitív infokommunikációs eszközök alkalmazása virtuális
kollaborációra</cim><kod>197</kod><oktatok>Baranyi Péter</oktatok><munkahely>BME
TMIT</munkahely><letszam>3</letszam><kivonat>A feladat célja olyan szoftver és hardver eszközök
fejlesztése, amelyek a virtuális világon keresztül lehetővé teszik emberek és robotok
kooperációját.</kivonat><reszletes>A jelentkező¶
```

13-3. ábra. XML szerű formátumban megadott forrásadatok

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Nézzünk most egy olyan feladatot, amit Word-makró írásával kell megoldanunk. Tegyük fel, hogy hallgatói témakiírások a 13-3. ábrán látható formában adottak. Egy-egy logikai egységet egy címke határoz meg. A címkék kezdetét a "<címke>" a végét a "</címke>" jelöli. Tegyük fel továbbá, hogy egy Word dokumentumban már mindegyik címkéhez létezik egy stílus, például a címhez az iw_cim nevű stílus, a létszámhoz az iw_letszam nevű stílus. A feladatunk ezután az, hogy a forrásadatok a 13-4. ábrán látható alakra hozzuk.

ERŐ ÉRZÉKELTETÉSE (FORCE FEEDBACK) KOGNITÍV INFOKOMMUNIKÁCIÓS CSATORNÁK ALKALMAZÁSÁVAL

Transmission of force feeling via cognitive infocommunication channels - 093

oktatók: Baranyi Péter
munkahely: BME TMIT
a témára 3 fő jelentkezését várjuk.
kivonat: A feladat célja olyan infokommunikációs algoritmusok fejlesztése, amelyek hatékony és költségkímélő mester-szolga telemanipulációt tesznek lehetővé.
leírás: A jelentkező feladata, hogy a nálunk kifejlesztett VirCA (Virtual Collaboration Arena) nevű 3D virtuális kognitív infokommunikációs platformhoz olyan szoftver komponenseket fejlesszen ki, amelyek lehetővé teszik a virtuális valóságban végzett telemanipuláció során fellépő, a valós manipulációban tapasztalható erők visszajelzését. A fejlesztésben lehetőség van japán, norvég és indiai hallgatókkal közösen dolgozni csoportmunkában.
előköv.: C#, C++ nyelvek ismerete, angol nyelvtudás, széleskörű érdeklődés és motiváltság.
kieg.: Kognitív infokommunikáció, Telemanipuláció, 3D virtuális környezet

KOGNITÍV INFOKOMMUNIKÁCIÓS ESZKÖZÖK ALKALMAZÁSA VIRTUÁLIS KOLLABORÁCIÓRA

Appling cognitive infocommunication devices for virtual collaboration - 094

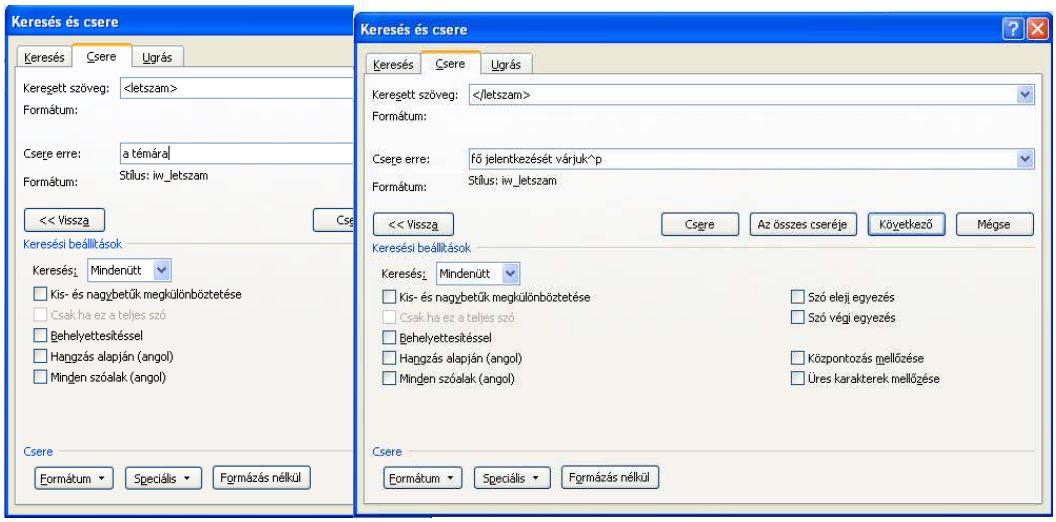
oktatók: Baranyi Péter
munkahely: BME TMIT
a témára 3 fő jelentkezését várjuk.
kivonat: A feladat célja olyan szoftver és hardver eszközök fejlesztése, amelyek a

13-4. ábra. A helyes formátumú témakiírások

Szerencsére különösebb programozási munka nélkül meg tudjuk ezt a feladatot oldani. Olyan makrókat kell rögzítenünk, ami a címkéket a megfelelő szövegrészletekre cseréli, miközben a címéhez tartózó stílust is beállítja. Nézzük meg példaként a 13-5. ábrát. A control-f gyorsbillentyű után elérhető Keresés és csere ablak Csere fülék az Egyebek gomb megnyomása után tudjuk a lecserélt szöveg stílusát beállítani az alsó sorban található Formátum gomb segítségével. Figyeljük meg, hogy a lecserélt szöveg formátuma iw_letszam lesz. A címke záró részénél a lecserélt szöveget a "^p" jelölés zárja. Ezt a speciális gomb megnyomása utáni menüből a bekezdésjel kiválasztásával tudjuk beszúrni. Sajnos a listában kétszer szerepel a bekezdésjel és csak a felső fog új sort beszúrni a cserék után.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



13-5. ábra. A létszám címke megformázása két cserével

Érdemes mindegyik címke cseréjére külön makrót rögzíteni és a sok kis makrót egy végső makróból sorban meghívni. Így ha az egyik címke cseréjének rögzítésekor hibázunk, akkor elég csak egy makró felvételét megismételni.

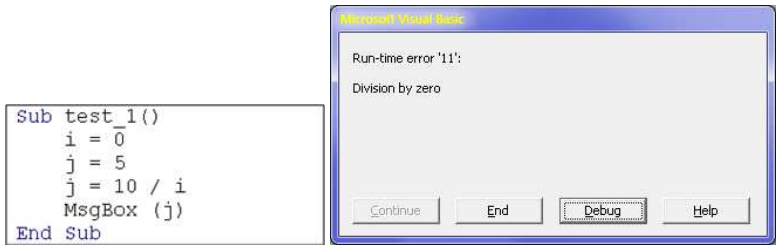
A Word makró elkészítése során láthattuk, hogy a Word programozása az Excelhez hasonlóan történik, ugyanazt a Basic programozási nyelvet használjuk, csak a két program programozási interfésze tér el egymástól, de a követendő fejlesztési elvek hasonlóak.

14. Futási hiba kezelése

Az On Error… nyelvi konstrukció fajtáji. További Basic nyelvi elemek: For Each, paraméterátadás, függvények. Rekurzió.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



14-1. ábra. Egyszerű példa a futási hibára

Nézzük meg a 14-1. ábra bal oldalán található egyszerű eljárást. Az i és a j változóknak az első két sorban értéket adunk, majd a harmadikban a j változónak újra értéket adunk, ezúttal egy, az i-t tartalmazó egyszerű számításon keresztül. Végezetül egy felugró ablakban tájékoztatjuk a felhasználót a j változó aktuális értékéről. A test_1 makró futtatásakor azonban a harmadik sor végrehajtásakor az ábra jobb oldalán látható ablakot dobja fel a Basic értelmező, mivel az i értéke nulla és a nullával való osztás futási hibát generált. Egy olyan hibába ütköztünk tehát, amit a Basic értelmező a programunk elindítása előtt nem vett észre, hiszen a "j = 10 / i" parancs akár helyesen le is futhatott volna, ha az i változóban nullától eltérő számot tároltunk volna.

Ha a Basic értelmező futási hibába ütközik, akkor végrehajtja az aktuális hibakezelő eljárását. Az alapértelmezett hibakezelő eljárás a 14-1. ábrán látható felugró ablakkal megszakítja a program futtását. Az aktuális hibakezelő eljárást az On Error kezdetű parancsokkal állíthatjuk be.

```
Sub test_2()
    i = 0
    j = 5
    On Error GoTo hiba
        j = 10 / i
    On Error GoTo 0
    MsgBox (j)
    Exit Sub

hiba:
    j = 2
    i = 1
    Resume Next
End Sub
```

```
Sub test_3()
    i = 0
    j = 5
    On Error GoTo hiba
        j = 10 / i
    On Error GoTo 0
    MsgBox (j)
    Exit Sub

hiba:
    j = 2
    i = 1
    Resume
End Sub
```

```
Sub test_4()
    i = 0
    j = 5
    On Error Resume Next
        j = 10 / i
    On Error GoTo 0
    MsgBox (j)
End Sub
```

14-2. ábra. Példák a hibakezelő eljárások használatára

A 14-2. ábra test_2 eljárása azt a hibakezelő eljárást mutatja be, amelyik futási hiba során a Basic program futását egy megadott címkétől folytatja. Az "On Error GoTo hiba" sorral beállítjuk, hogy futási

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

hiba bekövetkeztekor a hiba címkétől kezdődő hibakezelési utasításokra ugorjon a Basic értelmező. A "j = 10 / i" sor végrehajtásakor be is következik a nullával való osztásból fakadó futási hiba, ezért a "hiba:" sorral megadott címkére ugrunk. Ezután a j és az i változóknak megváltoztatjuk az értékét és a "Resume Next" paranccsal a futási hibát kiváltó parancs utáni parancsra ugrunk vissza, azaz az "On Error GoTo 0" sorra. Ez a parancs megint átállítja a hibakezelő eljárást. A GoTo 0 nem a nullás címkére fog ugrani hiba esetén, ugyanis a 0 egy speciális címke, ami az alapértelmezett hibakezelési eljárásnak felel meg. Tehát az "On Error GoTo 0" parancs végrehajtása után egy futási hibába futnánk bele, akkor a 14-1. ábrán megismert felugró ablak jelenne meg. A test_2 eljárás következő sorában egy felugró ablakot jelenítettünk meg j értékével, ami jelen esetben kettő. Végezetül az "Exit Sub" paranccsal befejeződik a test_2 eljárás végrehajtása. Az "Exit Sub" parancsra azért van szükség, mert hiányában újra a hibakezelési parancsaink hajtódnának végre.

Nézzük most meg a test_3 eljárást. Az egyetlen különbség a test_2 eljáráshoz képest az, hogy a hibakezelő parancsaik végén nem "Resume Next" hanem "Resume" szerepel. A Resume visszaugrik a hibát kiváltó parancs végrehajtására, így a hibát kiváltó parancs ismételten végrehajtódik. Jelen példánkban ez azt jelenti, hogy j értéke 10 lesz és a felugró ablakban ez a szám fog megjelenni.

A test_4 esetén a hibakezelési eljárást nem egy címkére ugrásra módosítjuk, hanem az "On Error Resume Next" paranccsal azt állítjuk be, hogy futási hiba esetén az eljárás végrehajtása a hibát kiváltó parancsot követő paranccsal folytatódjon. Emiatt a test_4 eljárás eredményeképpen a felugró ablakban a "5" szöveget fogjuk olvasni.

Nézzük meg most a 14-3. ábra test_5 nevű eljárását. Mi fog történni ennek az eljárásnak a futtatása során? Nem fogunk semmilyen felugró ablakot látni. A nullával osztási hiba után fellépő hibakezelési eljárás a hiba címétől folytatja az eljárás végrehajtását, azonban a Resume vagy Resume Next parancsok hiánya miatt az "i = 1" parancs végrehajtás után következő "End Sub" sorral véget ér az eljárás végrehajtása.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

```
Sub test_5()
    i = 0
    j = 5
    On Error GoTo hiba
        j = 10 / i
    On Error GoTo 0
    MsgBox (j)
    Exit Sub

hiba:
    j = 2
    i = 1
End Sub
```

```
Sub test_6()
    i = 0
    j = 5
    If i = 0 Then
        'hiba
        MsgBox "nullával nem osztunk"
    Else
        j = 10 / i
    End If
    MsgBox (j)
End Sub
```

14-3. ábra. További példák a hibakezelési módszerekre

Az eddigi példákban mindig a nullával való osztás miatt fellépő futási hibát kezeltük le így vagy úgy. Azonban a teljesség kedvéért meg kell említeni, hogy ez pont egy olyan futási hiba, amit teljesen is el tudunk kerülni egy egyszerű feltételvizsgálattal. Erre példa a test_6 nevű eljárás. Az Excel programozása során azonban gyakran találkozunk olyan lehetséges futási hibákkal, amit jóval egyszerűbb a hibakezelési eljárás beállításával kiküszöbölnünk.

Nézzük meg még egyszer a test_2 eljárást. A saját, speciális hibakezelési eljárásunk pusztán egyetlen sor végrehajtása során volt aktív, előtte és után az alapértelmezett hibakezelési eljárás volt érvényben. Ez ugyan nem kötelező, érdemes a nem alapértelmezett hibakezelést minél rövidebb időre korlátozni: arra a néhány utasításra, amelynek során arra számítunk, hogy esetleg olyan futási hiba fog fellépni, amely speciális kezelést igényel. Ezeket a kódrészleteket, az "On Error" közti rövid részeket, érdemes beljebb is szedni, hogy rögtön szembetűnő legyen, hogy ott speciális hibakezelés van érvényben.

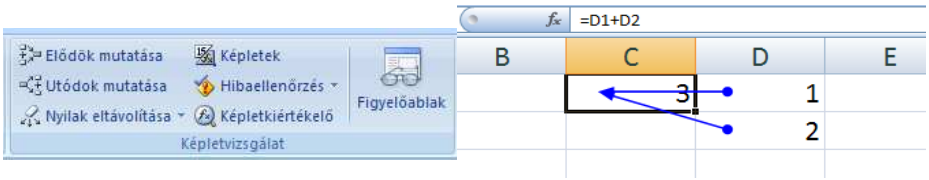
15. Függőségek feltérképezése

Elődök, utódok feltárása a felhasználói interfészek keresztül, számítási beállítások. Elődök, utódok feltárása Visual Basicben

Egy-egy nagyobb munkalap sok olyan formulát tartalmazhat, amely sok másik cellára, tartományra hivatkozik, amelyek szétszórva helyezkednek el a munkalapon, vagy esetleg más munkalapokon találhatóak. Mi magunk is azt az elvet követtük a 7. fejezetben, hogy egy bonyolultabb formula helyett sok kicsi formulából építkezve állítsuk össze egy-egy bonyolultabb feladatra adott válaszunkat. Ez azonban azzal jár, hogy utólag nem egyszerű átlátni, hogy melyik cella melyik másik cellától függ, illetve, hogy melyik másik cellára van hatással.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

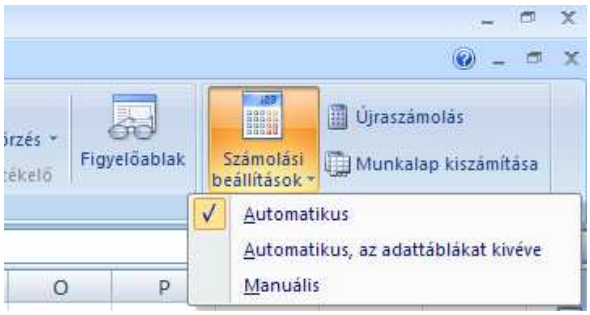
MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



15-1. ábra. A Képlet szalag képletvizsgálat csoportja és egy minta az elődök feltárására

A Képletvizsgálat csoportban található Elődök és Utódok mutatása ikonok használat ilyen esetben nyújt nélkülözhetetlen segítséget (15-1. ábra). Például ha a C1-es cellában a "=D1+D2" formula szerepel és a C1-es cella az aktív, akkor az elődök mutatása funkció előhívásának hatására az ábra jobb oldalán látható nyilak fogják mutatni a formula függőségeit. Ha ismételten ráklikkelünk az Elődök mutatása ikonra, akkor a formula elődeinek az elődeit is bejelöli az Excel.

Egy-egy bonyolultabb formula miatt sok ideig tarthat egy-egy munkalap módosítás után a megváltozott forrásadatok által előidézett újraszámítása a formuláknak. Ráadásul egy újraszámított formula egy másik újraszámítását vonhatja maga után, ami egy dominószerű reakció beindítását is eredményezheti. Ha az Excel grafikus interfészen keresztüli normál használata során találkozunk ilyen jelenséggel, akkor érdemes manuálisra váltani a formulák újraszámolását (15-2. ábra). A programozási interfészen keresztül a számítási beállítást a "Application.Calculation = xlCalculationManual" paranccsal tudjuk kézire váltani, "Application.Calculation = xlAutomatic" paranccsal automatikusra váltani. Ha manuális módban vagyunk, akkor "Application.Calculate" paranccsal tudjuk az összes munkafüzet újraszámítását előidézni.



15-2. ábra. A Képeltek szalag Számítás csoportja

Az elődök és utódok feltérképezésére a programozási interfészen keresztül is van lehetőségünk. Könnyen tudunk olyan makrókat írni, amelyek figyelembe veszik egy formula függőségi viszonyait. Ha

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

egy t egy tartomány-kifejezés, akkor a t.Precedents a t összes elődét tartalmazó, a t.Dependents pedig a t összes utódját tartalmazó tartomány-kifejezés lesz. A t.DirectPrecendents csak a közvetlen elődöket, míg a t.DirectDependents csak a közvetlen utódokat tartalmazza.

Ezen a ponton érdemes beírnunk az egyik kifejezést a Visual Basic Editorba, mondjuk így: ActiveCell.Precedents és Precenents szón hagyva a kurzort az Control-F1 gyorsbillentyűt lenyomni. Ennek eredményeképpen előugrik az Editor környezetérzékeny súgója. Szánjunk néhány pillanatot az angol nyelvű súgó tanulmányozására. Sok esetben lesz hasznos, ha részletes tanulmányozás után pontosan megértjük, hogy mi egy-egy eljárás pontos definíciója.

A képletvizsgálatokhoz használható programozási interfészen keresztül tudnánk olyan eljárást írni, ami attól függően állítaná be a cellák háttérszínét, hogy hányadfokú elődei, vagy utódai az aktív cellának. Ez az eljárás az alapértelmezett nyilaknál átláthatóbb felületet biztosítana a függőségi hierarchia megéretéséhez. De ezen túl is számos gyakorlat példát lehetne hozni, ami a képletek függőségeinek feltárására épülne. Most mégis inkább nézzünk egy egyszerű tanfeladatot. Írjunk egy makrót, ami kiszámítja egy cella összes elődcellájának összegét.

```
Sub my_sum()
    Range("f1") = 0

    For Each g In ActiveCell.Precedents
        Range("f1") = Range("f1") + g
    Next
End Sub
```

15-3. ábra. Markó, ami kiszámítja az aktív cella elődcelláinak összegét

A megoldás forráskódja a 15-3. ábrán olvasható. Használhatunk volna segédváltozókat is és közölhettük volna a felhasználóval a végeredményt egy felugró ablakban, de most a számítás részeredményeit és a végeredményt az F1-es cellában tároljuk. A megoldásban az egyetlen újdonság a "For Each" ciklus. Ez a for ciklustól eltérően nem egész számokon iterál, hanem az In kulcsszó után megadott gyűjteményen. Gyűjtemény sok minden lehet, de nekünk most legyen elég annyi, hogy tartomány-kifejezéseket is használhatunk a For Each ciklus ezen részében. A g nevű ciklusváltozó a ciklusmag mindenegyes lefuttatásakor a megadott tartomány-kifejezés egy-egy cellájára fog mutatni. A ciklusmag ezek után már könnyen értelmezhető: a részeredményekhez egyesével hozzáadjuk az elődök értékét.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

16. Munkalapok kezelése Visual Basicből

ActiveSheet, Select, Add, Name, iteráció a munkalapokon

A felhasználói interfészen keresztül könnyen tudunk új munkalapot hozzáadni egy munkafüzethez. Meg tudjuk változtatni egy munkalap nevét, esetleg átrendezhetjük a munkalapok sorrendjét. Ugyanezeket a feladatokat a programozási interfészen keresztül is elvégezhetjük.

Nézzük meg tehát milyen programozási interfészt biztosít az Excel a munkalapok manipulálására. A Sheets objektumváltozó egy olyan gyűjtemény, ami az aktuális munkafüzet munkalap-objektumait tartalmazza. A Sheets(1).Select parancs az első munkalapot választja ki, azaz teszi aktívvá. Ezzel szemben a Sheets("1").Select parancs az 1 nevű munkalapot aktiválja. A Set sh = Sheets(1) parancs hatására sh változó értéke egy referencia lesz az első munkalapra. A Set sh = Sheets.Add parancs hatására egy új, üres munkalap jön létre és adódik hozzá a munkafüzethez, az sh változó erre az új objektumra fog mutatni.

Az ActiveCell változóhoz hasonlóan az ActiveSheet az aktív munkafüzet előtérben lévő, azaz aktív munkalapjára mutató változó.

```
For Each sh In Sheets
    MsgBox sh.Name
Next
```

16-1. ábra. Kódrészlet a munkalapok neveinek kiíratására

Ha sh egy munkalapra mutató objektum, akkor az sh.Name kifejezés a munkalap nevét adja vissza. Mivel a Sheets egy gyűjtemény, ezért a Sheets objektumban szereplő munkalapokon egy For Each ciklussal végig iterálhatunk. Erre mutat példát a 16-1. ábrán látható kódrészlet. A munkafüzetben szereplő munkalapok neveit egyenként egy felugró ablakban közöljük a felhasználóval.

Az eddig használt tartomány-kifejezéseink mindig az aktuális, azaz az aktív munkalapra vonatkozott. Ha sh egy munkalapra mutató objektumváltozó, akkor sh-koz képesti tartomány-kifejezéseket is létrehozhatunk. Tehát az ActiveCell az aktuális munkalapon lévő aktív cellára mutató tartomány-kifejezés, míg az sh.ActiveCell az sh munkalapon lévő aktív cellára mutat. Hasonlóan az sh.Range("F3") az sh munkalap F3-as cellájára mutat.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

A munkalapok használatára következő fejezetben látunk részletesebb példákat. Most csak nézzük a 16-2. ábrán látható Basic eljárás kódját. Az öt sorból álló eljárás egyetlen lényegi parancsot tartalmaz: a Sheets(1).Delete kitörli az első munkalapot. A kitörlés azonban adatvesztéssel járhat, ezért az Excel egy felugró ablakban kér megerősítést a felhasználótól a törlési művelet végrehajtása előtt. A felhasználói megerősítést úgy tudjuk elkerülni, hogy a Application.DisplayAlerts változó értékét hamisra állítjuk. Végezetül ha a munkafüzet egyetlen egy munkalapból áll, akkor a törlés nem engedélyezett és a törlési kísérlet futási hibát eredményez. A futási hiba elrejtést oldja meg a 14. fejezetben megismert hibakezelési módszer.

```
Sub del()
    Application.DisplayAlerts = False
    On Error Resume Next
        Sheets(1).Delete
    On Error GoTo 0
    Application.DisplayAlerts = True
End Sub
```

16-2. ábra. A munkafüzet első munkalapjának eltávolítását megvalósító eljárás

17. Adat importálása webről

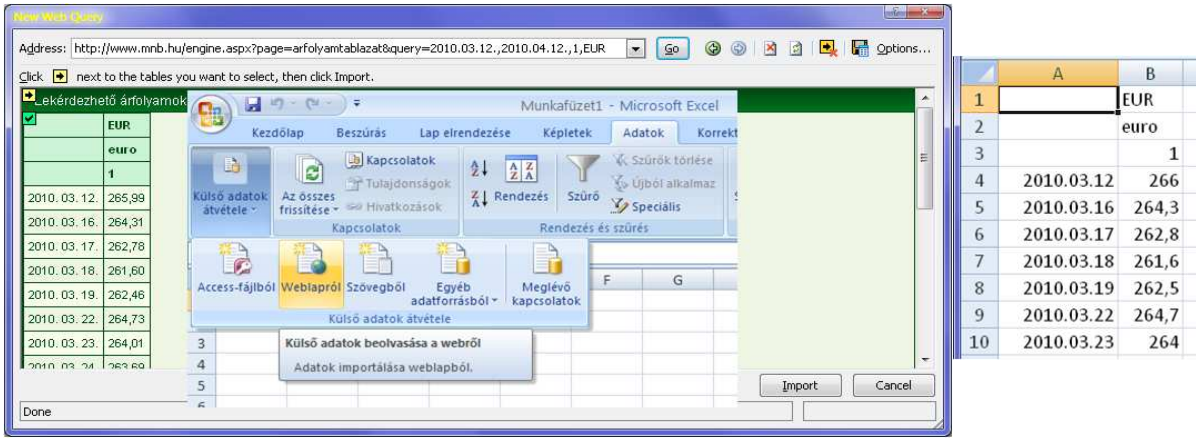
Stringek összefűzése. Adatok importálása a felhasználói és a programozói interfészen keresztül. . Grafikonok automatikus előállítása

Excel munkalapokra felvihetünk mi magunk adatokat, amelyet aztán különböző módon értékelhetünk. A 4. fejezetben láthattuk, hogy készíthetünk olyan űrlapként funkcionáló munkafüzeteket, amelyeket aztán munkatársaink között terjesztve ők tudják az általunk meghatározott formátumú adatokkal feltölteni a munkafüzetünket. Az Excel fájl menüjét megnézve láthatjuk azt is, hogy megnyithatunk szöveges fájlokat és xml formátumban tárolt adatokat is. Sőt, adatbázisból fel tudunk tölteni egy munkalapot.

Az Adatok szalag Külső adatok átvétele nevű csoportjában meglévő munkalapba tudunk adatokat importálni. A 17-1. ábra azt mutatja, hogy hogyan lehet az MNB weboldaláról betölteni egy munkalapba a forint-euró árfolyamot. Miután az ábra közepén látható Adatok szalagról kiválasztottuk a Weblapról történő adatimportálást az ábra hátterében látható, egy jól ismert webböngészőhöz hasonló böngészőablak jelenik meg. Ebben a böngészőben kell az importálandó adatokat tartalmazó weboldalra navigálnunk. A böngésző abban különbözik a megszokott böngészőktől, hogy a weboldalakon található

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

táblázatok bal felső sarkába egy sárga hátterű nyilat helyez el. A sárga hátterű nyilakra klikkelve tudjuk kijelölni, hogy az oldalon található táblázatok közül melyeket szeretnék betölteni az Excelbe. A kijelölt táblázatok bal felső sarkában zöld hátterű pipa fog megjelenni.



17-1. ábra. Árfolyamadatok importálása a Magyar Nemzeti Bank weboldaláról

A böngésző jobb alsó sarkában lévő Import, vagy a magyar nyelvű változatban az Importálás gombra klikkelve tudjuk az adatbetöltést kezdeményezni. Az importálás eredménye az 17-1. ábra jobb oldalán látható.

Az webről történő adatimportálás egyik nagy előnye, hogy az adatok nem egyszerűen csak bemásolódnak a munkalapra, az Excel azt is eltárolja a munkalapon, hogy melyik weboldal hányadik táblázatát, illetve táblázatait jelöltük ki. Ezért például ha a harmincnapos időjárási előrejelzés hőmérsékleti adatai importáljuk be egy meteorológia weboldalról, akkor az Excel munkalap frissítése az aktuális adatok újbóli letöltését is eredményezi.

A webes importálás azonban csak akkor működik, ha az importálandó adathoz létezik közvetlen webcím és ha az adat táblázatos formában van tárolva. Sok olyan dinamikus tartalom érhető el a weben, aminél a weboldal tartalma úgy változik meg, hogy az oldalhoz tartozó webcím változatlan marad. Továbbá az utóbbi években elterjedt webdesign a táblázatos adatokat nem az Excel által elvárt táblázatban jeleníti meg.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

A fentiek miatt a webes import használhatósága némileg korlátozott, azonban így számos esetben hasznos tud lenni. Lássunk egy példát. A boxofficemojo.com oldal az Amerikában bemutatott mozifilmek bevételi adatait tesz elérhetővé. A filmek különböző adatait tartalmazó weboldalakhoz szerencsére különböző webcím tartozik. A filmek címe azonban nem közvetlenül szerepel a webcímben, hanem egy, a filmcímből generált azonosítón keresztül. Például a "Bad Teacher" című film azonosítója a "badteacher" és a film heti bontásban elért bevételeit tartalmazó oldal webcím a következő: http://boxofficemojo.com/movies/?page=weekly&id=badteacher.htm. Hasonlóan a "Harry Potter and the Deathly Hallows, Part 2" című film azonosítója a "harrypotter72" kulcsszó és a heti bontásban elérhető teljesítménye a filmnek a következő címen olvasható: http://boxofficemojo.com/movies/?page=weekly&id=harrypotter72.htm

Írjuk ezután egy olyan Basic eljárást, ami először is az aktuális cellában lévő filmazonosítóhoz létrehoz egy munkalapot az azonosítóval megegyező névvel. Másodszor, az új munkalapba eltölti a film heti bontásban elért eredményét. Harmadszor, a film eredményét grafikus formában ábrázolja. Végezetül pedig az aktív cellát a makró meghívása előtti állapotba állítja vissza.

Elsőként az aktuális cellában lévő filmazonosítóval megegyező nevű munkalapot kell létrehoznunk. A 17-2. ábrán látható kódrészletben először az id nevű változóban eltároljuk az aktív cellában lévő filmazonosítót. Majd a currentSh változóban az aktuális munkalapra mutató objektumot. A harmadik sorban pedig egy új munkalapot adunk a munkafüzethez és az új munkalapra mutató objektumot az sh változóban tároljuk el. Ezek után az sh munkalap nevét a filmazonosítóra cseréljük. Végezetül az új munkalapot az előtérbe hozzuk. Vegyük észre, hogy az id egy hagyományos változó, amiben string típusú értéket tároltunk, míg a currentSh és az sh objektumváltozó, amiben egy munkalapra mutató objektumot tároltunk. Emiatt kellett az utóbbi két esetben Set paranccsal kezdeni az értékadást.

```
id = ActiveCell.Value

Set currentSh = ActiveSheet
Set sh = Sheets.Add
sh.Name = id
sh.Select
```

17-2. ábra. Kódrészlet az első részfeladathoz

Második részfeladatként az új munkalapra le kell tölteni a filmhez tartozó heti bevételi eredményeket. Vajon hogyan kell erre programot írni? Próbálkozhatnánk a Súgó átbogarászásával, de valószínűleg igen sokára jutnánk használható információhoz. Jobban járunk, ha rögzítünk egy makrót, ami az egyik filmhez

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

elvégzi az adatimportálást. A rögzített makró forráskódja (17-3. ábra) elsőre nem túl sokat mondó, de nem is az a célunk, hogy pontosan milyen utasításokat rögzítettünk. A célunk az, hogy valahogy kitaláljuk, hogy hogyan kell ezt a makrót úgy módosítani, hogy ne csak a Bad Teacher című film letöltését vezérelje, hanem tetszőleges filmét. A forráskód tüzetesebb tanulmányozása után észrevehetjük, hogy a film azonosítója kétszer is szerepel a kódban. Az első előforduláskor ráadásul még felfedezhetjük a bevételek heti bontását mutató weboldal webcímét is. Nincs tehát más dolgunk, mint a film azonosítóját tartalmazó stringeket úgy módosítani, hogy az id változó értéke szerepeljen benne a badteacher helyett.

Ha s1 és s2 egy-egy string kifejezés, akkor az "s1 & s2" egy olyan kifejezés, ami a string összefűzéséből adódik. Tehát a "mozi" & "film" eredménye a "mozifilm" lesz. Így már érhető a 17-4. ábrán látható második részfeladatra adott programrészlet. A webcim és a nev változókban eltároljuk a filmazonosítókból generált stringeket és ezeket használjuk fel az importálási parancsok argumentumaiban.

```
Sub badteacher_import()
    With ActiveSheet.QueryTables.Add(Connection:= _
        "URL;http://boxofficemojo.com/movies/?page=weekly&id=badteacher.htm", Destination _
        :=Range("$A$1"))
        .Name = "?page=weekly&id=badteacher"
        .FieldNames = True
        .RowNumbers = False
        .FillAdjacentFormulas = False
        .PreserveFormatting = True
        .RefreshOnFileOpen = False
        .BackgroundQuery = True
        .RefreshStyle = xlInsertDeleteCells
        .SavePassword = False
        .SaveData = True
        .AdjustColumnWidth = True
        .RefreshPeriod = 0
        .WebSelectionType = xlSpecifiedTables
        .WebFormatting = xlWebFormattingNone
        .WebTables = "7"
        .WebPreFormattedTextToColumns = True
        .WebConsecutiveDelimitersAsOne = True
        .WebSingleBlockTextImport = False
        .WebDisableDateRecognition = False
        .WebDisableRedirections = False
        .Refresh BackgroundQuery:=False
    End With
End Sub
```

17-3. ábra. A Bad Teach című film importálása során rögzített makró (egyszerűsített) forráskódja

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

```
webcim = "URL;http://boxofficemojo.com/movies/?page=weekly&id=" & id & ".htm"
nev = "?page=weekly&id=" & id

With ActiveSheet.QueryTables.Add(Connection:= _
    webcim, Destination _
    :=Range("$A$1"))
    .Name = nev
    .FieldNames = True
    .RowNumbers = False
    .FillAdjacentFormulas = False
    .PreserveFormatting = True
    .RefreshOnFileOpen = False
    .BackgroundQuery = True
    .RefreshStyle = xlInsertDeleteCells
    .SavePassword = False
    .SaveData = True
    .AdjustColumnWidth = True
    .RefreshPeriod = 0
    .WebSelectionType = xlSpecifiedTables
    .WebFormatting = xlWebFormattingNone
    .WebTables = "7"
    .WebPreFormattedTextToColumns = True
    .WebConsecutiveDelimitersAsOne = True
    .WebSingleBlockTextImport = False
    .WebDisableDateRecognition = False
    .WebDisableRedirections = False
    .Refresh BackgroundQuery:=False
End With
```

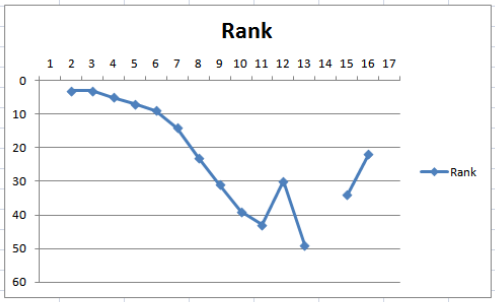
17-4. ábra. Az id váltózóban tárolt filmazonosítóhoz kapcsolódó adatok importálása (programrészlet)

A harmadik részfeladat a letöltött adatok alapján egy grafikon automatikus előállítása. Az előzőekben használt módszerrel próbálkozunk most is. Rögzítünk egy makrót, ami a Bad Teacher film letöltött adataiból (17-5. ábra bal oldala) előállítja a kérdéses grafikont (17-5. ábra jobb oldala) és megnézzük, hogy mit kell rajta módosítani, hogy általánosan használható makrót kapjunk.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

	A	B	C	D	E	F	G	H	I
1	Date	Rank	Weekly	%	Theaters / Change		Avg.	GrosstoDate	Week
2	(click to view chart)		Gross	Change					#
3	Jun 24–30	3	$45,445,971		3,049		$14,905	$45,445,971	1
4	Jul 1–7	3	$24,311,436	46,50%	3,049		$7,974	$69,757,407	2
5	Jul 8–14	5	$13,547,785	44,30%	2,962	87	$4,574	$83,305,192	3
6	Jul 15–21	7	$8,449,571	37,60%	2,659	303	$3,178	$91,754,763	4
7	Jul 22–28	9	$4,120,644	51,20%	2,034	625	$2,026	$95,875,407	5
8	Jul 29–Aug 4	14	$1,296,735	68,50%	765	1,269	$1,695	$97,172,142	6
9	Aug 5–11	23	$525,452	59,50%	314	451	$1,673	$97,697,594	7
10	Aug 12–18	31	$304,642	42,00%	149	165	$2,045	$98,002,236	8
11	Aug 19–25	39	$162,244	46,70%	100	49	$1,622	$98,164,480	9
12	Aug 26–Sep 1	43	$135,908	16,20%	75	25	$1,812	$98,300,388	10
13	Sep 2–8	30	$679,661	400%	986	911	$689	$98,980,049	11
14	Sep 9–15	49	$84,300	87,60%	96	890	$878	$99,064,349	12
15	Sep 16–22		$459,817	446%	474	378	$970	$99,524,166	13
16	Sep 23–29	34	$228,032	50,40%	232	242	$983	$99,752,198	14
17	Sep 30–Oct 6	22	$508,104	123%	354	122	$1,435	$100,260,302	15
18	Oct 7–13		$32,554	93,60%	64	290	$509	$100,292,856	16



17-5. ábra. A Bad Teacher című filmhez importált adatok és az azokból előállított grafikon

Figyeljük meg, hogy a grafikonon szakadások találhatók. Ez abból adódik, hogy a 15. és a 18. sor B oszlopában nincs adat, mert valószínűleg ezekben a hetekben a film nem szerepelt az első ötven legtöbb bevételt produkáló filmje között.

```
Sub grafikon()
    Columns("B:B").Select
    ActiveSheet.Shapes.AddChart.Select
    ActiveChart.SetSourceData Source:=Range("$B:$B")
    ActiveChart.ChartType = xlLineMarkers
    ActiveChart.Axes(xlValue).Select
    ActiveSheet.ChartObjects("Chart 1").Activate
    ActiveChart.Axes(xlValue).ReversePlotOrder = True
    Range("A1").Select
End Sub
```

17-6. ábra. A grafikon generálásához rögzített makró forráskódja

A rögzített makró forráskódja a 17-6. ábrán látható. Mivel a grafikon forrásadatának a teljes B oszlop volt megadva (harmadik sor) ezért a grafikon-generáló eljárás nem függ attól, hogy az adott filmet hány hétig vetítették az Amerikai mozik. Így a grafikon makró változatlanul felhasználható a harmadik részfeladat megoldásához. El kell azonban mondani, hogy a grafikon makró korántsem univerzális. Az eljárás ugyanis feltételezi, hogy az eljárás meghívása előtt az aktív munkalap sohasem tartalmazott grafikont, mivel az újonnan létrejött grafikonra "Chart 1" néven hivatkozik.

Végezetül az utolsó részfeladat, hogy állítsuk az aktív cellát az eredeti állapotába, tehát váltsunk vissza a kiindulási munkalapra. Erre a munkalapra mutató referenciát a currentSh változóban már korábban eltároltunk, így ez a részfeladat a currentSh.Select paranccsal megoldható. Összefoglalva tehát a

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

A projektek az Európai Unió támogatásával valósulnak meg.

teljesen feladat megoldható a 17-2., a 17-4., a 17-7. ábrán található kódrészletek összefűzésével és a 17-6. ábrán látható eljárással.

```
grafikon
currentSh.Select
```

17-7. ábra. A feladat megoldásának utolsó kódrészlete

Legvégül azonban meg kell jegyezni, hogy a megírt eljárásunk sok filmazonosítóval helyesen fog működni, de sajnos nem az összessel. Ha jobban megnézzük a 17-3. ábrát, akkor észrevehetjük, hogy a WebTables értéke "7". Ez azt jelenti, hogy az webes importálás során a weboldal hetedik táblázatát jelöljük ki importálásra. Sajnos azonban a boxofficemojo több táblázatban adja meg a heti bevételeket, ha a filmet több évben is vetítették. Tehát például akkor, ha a filmet karácsony előtt mutatták be és még a következő év januárjában is a mozik műsorán volt. A programunk természetesen módosítható lenne úgy, hogy ilyen esetekben is helyesen működjön. A módosítás azonban didaktikai szempontból felesleges elbonyolítaná a programot.

18. Makrók haladó használata

Makró hozzárendelése nyomógombhoz. Szalagok illetve a gyorselérési eszköztár testreszabása. Saját munkalap-függvény írása

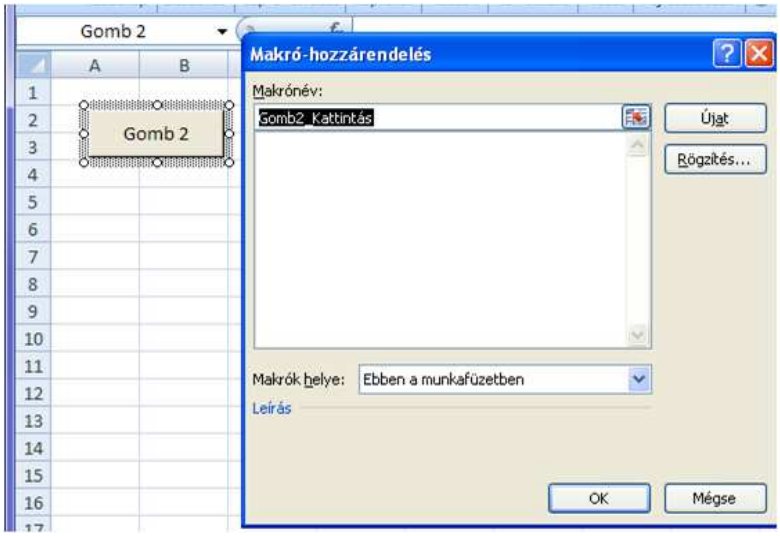
Makrókat létrehozhatunk makrórögzítés segítségével, egy korábban rögzített makró módosításával, vagy egyszerűen egy argumentum nélküli Basic eljárás megírásával. Mivel az általunk eddig írt makrók nem használtak paraméterátadást azaz nem volt argumentumuk, ezért ezek mind megjelentek a futtatható makrók listájában. Minden makróhoz nem rendelhetünk gyorsbillentyűt, mert előbb-utóbb az általunk is gyakran használt beépített gyorsbillentyűket lennénk kénytelenek felüldefiniálni. Néha viszont a futtatható makrók listájában is macerás az a konkrét makrót, amit szeretnénk visszajátszani. Sokszor még az is kizökkenhet a munkamenetünkből, ha az F8 gyorsbillentyűt használjuk és nem a nem is a fejlesztőeszközök szalagról hozzuk be a makrók listáját.

Makrók előkeresését, illetve futtatásának elkezdését gyorsíthatjuk úgy is, hogy egy nyomógombot helyezzük el a munkalapon, amire klikkelve egy előre meghatározott makró futtatását kezdeményezzük. Az így létrehozható nyomógomb is a 4. fejezetben megismert űrlap-vezérlőelemek egyike. A vezérlőelem beillesztésekor a 18-1. ábrán látható ablak jelenik meg. Itt a vezérlőelem csatolását állíthatjuk be. Nyomógomb esetén azonban a vezérlőelemet nem egy cellához kell rendelni, hanem egy

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

makróhoz. Új makrót hozhatunk létre, vagy akár rögzíthetünk is egyet. Illetve a létező makróink közül választhatunk ki egyet. Későbbiekben a nyomógombot jobb egérkattintással jelölhetjük ki és módosíthatjuk a csatolását. Ha a nyomógomb ki van jelölve, akkor a balklikk után szerkeszthetjük a nyomógomb feliratát.

Nyomógombokhoz rendelt makrókat igazán akkor érdemes használnunk, ha a makrónk nem univerzális. Azaz ha a makrónak a nyomógomb környékén van értelme. Egy makró többé-kevésbé univerzális, ha sok munkalapon, vagy sok cellából indítva értelmes műveletsort hajt végre. Egyszerű univerzális makróra lehet példa egy olyan makró, ami az aktív cella háttérszínét a cella szomszédcelláinak éréke alapján határozza meg. Egy konkrét feladatra használható makróra lehetne példa olyan eljárás, ami a 3-1. ábrán látható véralkoholszint-kiszámító táblázatnak kinullázza a fogyasztott mennyiségeket tartalmazó oszlopát.



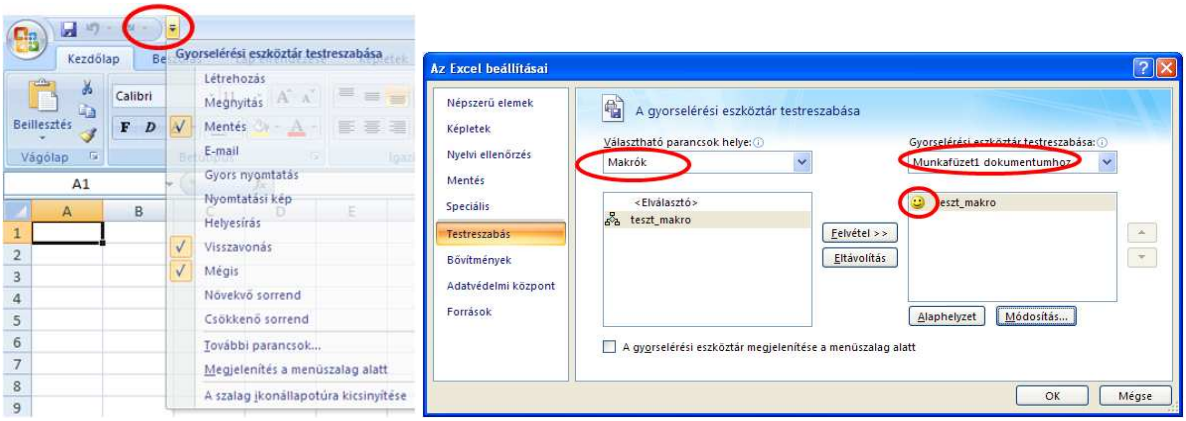
18-1. ábra. Nyomógomb vezérlőelem létrehozása és makróhoz való rendelése

Van lehetőség arra is, hogy egy munkafüzethez egyéni szalagokat adjunk hozzá, vagy hogy a beépített szalagokat szabjuk testre. Ezekhez a beállítási lehetőségekhez azonban nem lehet hozzáférni a felhasználói interfészen keresztül. Sőt, a Basic programozási interfészen keresztül sincs erre lehetőség. A szalagot módosítását csak a Basicnél jóval bonyolultabb VSTO programozási interfész használata teszt

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

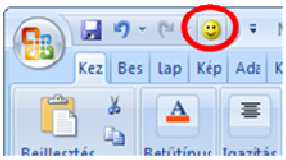
MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

lehetővé. Illetve a szalag minimális módosítását az Excel, Word fájlok bonyolult kézi szerkesztésével is el lehet érni. Ezeket a megoldások azonban itt nem tárgyaljuk.



18-2. ábra. A gyorselérési eszköztár testreszabásának lépései

Van lehetőségünk viszont az Excel, Word, PowerPoint bal felső sarkában található gyorselérési eszköztár testreszabására. A 18-2. ábra bal oldalán látható További parancsokra jelenik meg a jobb oldalon látható részletes beállítási ablak. A választható parancsok helyénél lehet kiválasztani, hogy egy általunk írt makrót szeretnénk az gyorselérési eszköztárba felvenni. Ha a makrót az adott munkafüzetbe mentjük le, akkora a testreszabás változtatásait is a munkafüzetbe érdemes lementeni, ahogy azt az ábra jobb felső sarában található piros jelölés is mutatja. A Módosítás gombbal a makróhoz tartozó ikont állíthatjuk be és a makró megjelenített nevét. A 18-3. ábra az új makróval kibővített eszköztárat mutatja.



18-3. ábra. A módosított gyorselérési eszköztár

Az eddig létrehozott Basic eljárásaink függetlenül attól, hogy mi magunk írtuk azokat, vagy csak makróként rögzítettük azokat, csak arra nyújtottak lehetőséget, hogy a cellák tartalmát az eljárás futása során mosósítsuk. Ha forrásadatok megváltoztak és az eljárásunkkal kapott eredmények elavultak, akkor újra le kellett volna futtatunk az eljárásunkat, hogy frissítsük a számításaink eredményét. Jóllehet

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

a 11-3. ábra mutatott arra példát, hogy egy eljárás nem a cella értékét, hanem a cellában tárolt formulát módosította. Azonban Basic programból nehézkes összetettebb formulákat összeállítani. Ráadásul a Basic programból így létrehozott formulák nem lesznek nagyobb tudásúak a grafikus interfészen keresztül beírt formuláknál.

Szerencsére mi magunk is tudunk munkalapfüggvényeket létrehozni. Ezeket a munkalapfüggvényeket aztán ugyanúgy felhasználhatjuk a munkalapjaink formuláiban mint például a beépített MAX, SUM, HA függvényeket.

Basicben az eljárásokhoz hasonlóan kell függvényeket definiálni. A függvényeket a Sub helyett a Function kulcsszó vezeti be és az End Function zárja. A függvény visszatérési értéket egy speciális változóba kell tárolnunk, aminek a neve megegyezik a függvény nevével. A 18-4. ábrán látható egyszerű függvénynek két argumentuma van. A függvény visszatérési értéke a paraméterül kapott két szám összege.

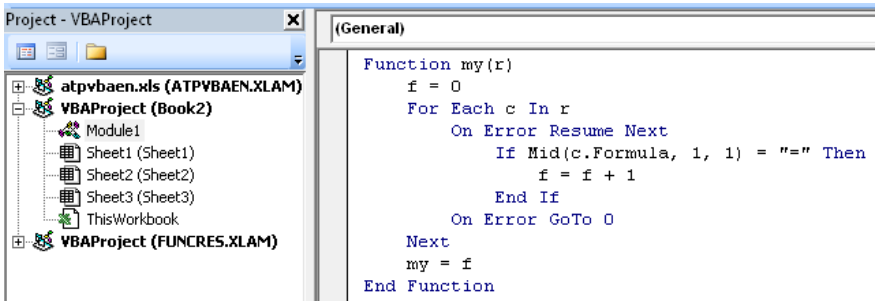
```
Function osszeadas(a, b)
    osszeadas = a + b
End Function
```

18-4. ábra. Egyszerű példa egy függvény deklarációra

A függvény egy modulban van definiálva, akkor felhasználható mint munkalapfüggvény. Erre mutat példát a 18-5. ábra. A my nevű függvény egyetlen argumentumot vár: egy tartomány-kifejezést. A az r tartomány minden cellájára megvizsgálja, hogy a cellában tárolt formula egyenlőségjellel kezdődik-e, ha igen az azt jelenti, hogy a c cellában valóban egy formula található, ekkor növeli az f váltózó értékét. Tehát a függvény az r tartományban található formulák számát adja vissza. Fontos, hogy a függvény egy modulban (a Module1-ben) található, mert csak így érhető el egy Excel munkalapról. Ha az A1-es cellában a "=my(B1:C1)" formula szerepel és a B1-es cella értéke 2 és a C1-ben az "=B1*2" formula található, akkor az A1-es cellában az 1 számot fogjuk látni.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



18-5. ábra. Basicben írt munkalapfüggvény

Saját munkalapfüggvények írása sokszor nagyon praktikus tud lenni, mert kényelmesen tesz elérhetővé speciális számítási függvényeket. Azonban a Basicben megírt munkalapfüggvényeknél különösen fontos, hogy hatékony, gyorsan lefutó kódot írjunk. Ezek függvények ugyanis mindent olyan cella frissítésekor lefutnak, aminek a formulája a függvényt is tartalmazza.

19. Előadás készítése

Előadás készítése PowerPointban, sablonok használata, animációk. Excel adatok bemutatása PowerPointban: adatok csatolási lehetőségei

A PowerPoint előadás-szerkesztő program annyira könnyen és intuitíven használható, hogy egy előadás fóliasorozatának elkészítésekor a nagyobb fejtörést a fóliák tartalmának kigondolás okozza és nem a kigondolt tartalom PowerPoint programban való megvalósítása. Érdemes azonban tisztában lenni a program képességeivel, hogy tudjuk a tartalmi tervezésnél, hogy milyen elemekből építkezetünk.

Az előadás kinézetét sablonokon keresztül tudjuk gyorsan megváltoztatni. Az előre elkészített sablonokon kívül, magunk is összegyűjthetjük a megjelenési elemek módosításai az úgy nevezett mintadia szerkesztésével. A diaminta szerkesztésére a Nézet szalag Bemutatónézetek csoportjának Diaminta ikonjával tudunk váltani.

A diákon található objektumokhoz animációt rendelhetünk. Egy animáció lehet megjelenési vagy eltűnési animáció is. Az animáció elindulhat automatikusan az előző animáció után, vagy az előadó egérklikkjére is. Ha nem ismerjük, érdemes egyszer végignézni az összes animálási opciót.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Tegyük most fel, hogy Excelben végzett számításainkból szeretnénk egy előadást készíteni, hogy munkánk főbb eredményit bemutassuk a főnökeinknek. Tegyük fel továbbá, hogy minden hónapban az adatok és a számítások aktualizálása után a munkánkat ismételten be kell mutatnunk. Több úton indulhatunk el.

Először is a bemutatni kívánt adatokat a vágólapon keresztül átmásolhatjuk az Excel munkalapról a PowerPoint diára. Ennek a megközelítésnek az a hátránya, hogy ha változik az Excelben tárolt forrásadat, akkor újra át kell másolnunk a megváltozott adatokat a diákra.

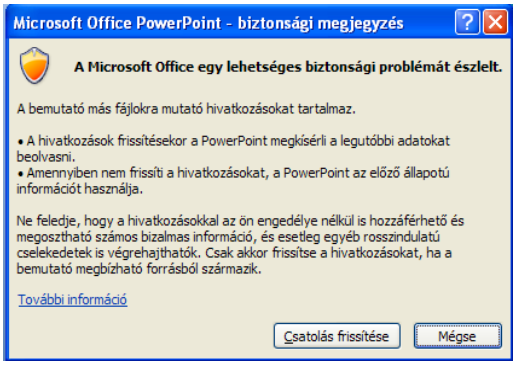
Második lehetőségként az egész Excel fájlt beágyazhatjuk az előadásba. Az adatok frissítését a beágyazott munkalapon végezhetjük el. Itt a nehézséget az jelenti, hogy hogyan tudunk ugyanabból a beágyazott dokumentumból két különböző helyen két érétket megjeleníteni.

Végezetül az Excel adatokat tudjuk csatolni (linkelni) az előadásunkhoz. Ebben az esetben, ha változik a forrásadat, akkor az előadás megnyitásakor lehetőségünk van az adatok frissítésére. Az Excelben helyezzünk a vágólapra egy cellát vagy egy grafikont, majd PowerPoint Kezdőlap szalagján válaszuk ki az irányított beillesztés csatolt változatát (19-1. ábra). Ha a későbbiekben újra megnyitjuk ezt az előadást, akkor a 19-2. ábrán látható felugró ablakon keresztül tudjuk egyetlen egy lépésben frissíteni az összes csatolt objektumot.

19-1. ábra. Az irányított beillesztés két lépése

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.



19-2. ábra. Csatolt adatok frissítése

20. Alternatív programcsomagok: OpenOffice.org / LibreOffice

A szabad szoftver definíciója. Lekötés (Lock-in); lehetséges védekezések; Office fájlformátumok. OpenOffice.org történelem. Dokumentumok on-line szerkesztése és tárolása a felhőben (Google Docs)

Az eddigi fejezetek csak rövid betekintést nyújtottak az Excel és a többi Microsoft Office program felhasználási lehetőségeibe, mégis látszik, hogy egy igen nagy tudású és sokrétű szoftvercsomagról van szó. A programcsomag használatának azonban vannak hátrányai. Ezek a hátrányok nem elsősorban a programcsomag hiányos funkcionalitásából adódik. Olvassuk csak bele a felhasználást szabályozó szoftverlicenc szerződésbe, amely több ponton korlátozza a felhasználót. A szerződés elolvasásának első lépeseként nyomjuk meg az Excel bal felső ablakában található Office gombot, majd az előugró menü alján található Excel beállításai gombot. Az Excel beállításai ablakban válasszuk ki a baloldali menüben található Források menüpontot, majd klikkeljünk a Névjegy ikonra. A névjegy panel közepén található Microsoft szoftverlicenc-szerződés megtekintése linkre klikkelve végül elolvasható a több mint hetven oldal hosszú szerződés.

A fenti licenccel szöges ellentétben állnak a nyolcvanas évek elején indult szabadszoftver-mozgalom alapelvei. A szabad szoftver Richard Stalmann által kidolgozott definíciója szerint egy szoftver, akkor szabad szoftver (free software), ha megfelel a következő négy feltételnek. (0) A szoftver bármilyen célra felhasználható. (1) Lehetőség van a szoftver működésének szabad tanulmányozására és módosítására. (2) A szoftver szabadon terjesztető, továbbadható. (3) A felhasználónak lehetősége van a szoftver továbbfejlesztésére és a fejlesztés közreadására. A definícióból nem feltétlenül következik, de a szabad szoftverek szinte mindig ingyenesen le is tölthetőek.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Tegyük fel, hogy egy feladat ellátására létezik egy ingyenesen hozzáférhető szabad szoftver és egy viszonylag drágán beszerezhető, a felhasználási feltételeket korlátozó szoftver is. Hogyan lehetséges ebben az esetben nagyon sokan nem a szabad változatot használják? Ennek oka legtöbbször az úgynevezett lekötés vagy angolul a lock-in. Lekötésről akkor beszélünk, ha egy drága alternatíváról nem éri meg váltani az olcsóbb változatra, mert a váltás költsége meghaladja az olcsóbb alternatíva alacsonyabb beszerzési és üzemeltetési költségeiből adódó megtakarításokat.

Milyen költsége lehet egy szabad szoftverre való váltásnak? Nyilván a beszerzési költség a jelentős. Azonban például hiába alkalmas ugyanazon feladatok elvégzésére a két szoftver, ha a felhasználói felületük eltérő, akkor esetleg jelentős betanítási költséggel kell számolni. A másik tipikus átállási költség az adatmigrációból adódik. Ez sokszor jelentősebb kiadást jelent. A lekötést teljes mértékűvé is teheti, ha a drágább szoftverrel elmenet adatfájlokat a másikkal nem lehet megnyitni, vagy nem létezik az átalakítást elvégző konvertáló program.

Az adatformátumok miatti lekötést ellen legjobban használható védekezés a szabványosítás. A szabványosításhoz kormányzati szerepvállalás, vagy a piaci szereplők összefogása szükséges. Ha létrejön egy egységes, a felhasználók által igényelt adatformátum, akkor a szoftvergyártók kénytelenek a formátumot támogatni. A termékek közti átjárhatóság a lekötés megakadályozása piaci versenyt von maga után, ami az árak csökkenéséhez és innovatívabb termékek kifejlesztéséhez vezethet.

A lekötés ellen vétkezhetnek a potenciális vevők úgy is, hogy összefogva fejlesztenek egy szabad szoftvert vagy egy ilyen szoftver fejlesztését támogatják. Az összefogás miatt az egy szereplőre eső költségek csökkenek. A megfelelő szabadszoftver-licenc pedig garantálja, hogy a szoftver fejlesztése és maga a szoftver sem kerülhet konkurens vállalatok irányítása alá.

A Microsoft Office termékekről való átállásnak is az adatformátumokon keresztüli elkötés volt a legfőbb gátja. Mára két nagyom szabványcsalád létezik az irodai programcsomagok fájlformátumára. Az egyik az ODF, azaz a Open Document Format for Office Applications. A másik az OOXML, azaz a Office Open XML formátum. Mindkét formátumot más szabványosítási szervezetnél fogadták el.

Az ODF formátumot több program is támogatja (OpenOffice / LibreOffice, KOffice) és a Microsoft Office is valamennyire. Az ilyen fájlok esetén a fájlok kiterjesztéseire az .odt, .ods, .odp végződések használatosak.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Az OOXML formátumnak a Microsoft az elsődleges támogatója. A tipikusan használatos kiterjesztések a .docx, .xlsx, és a pptx.

Érdemes egy kicsit részletesebben is megismerni a legjelentősebb szabad szoftveres irodai programcsomag történetét, mert benne a szabad szoftver projektet sok tipikus vonatkozását megtaláljuk. A StarDivision céget 1999-ben felvásárolta a Sun Microsystems és a cég StarOffice termékéből létrehozta annak a szabad szoftveres változatát Openfoofice.Org néven. Az org végződés az volt hivatott jelenteni, hogy a programot egy nyílt közösség fejleszti. Bárki küldhetett be javításokat, illetve új programkódot a projekthez. Azonban a fő fejlesztő, támogató egyértelműen a Sun volt. A közösségi fejlesztés koránt sem volt zökkenőmentes, a külső fejlesztők és a Sun között sok súrlódás hátráltatta az együttműködést. 2010-ben aztán a Sunt felvásárolta az Oracle vállalat. A felvásárlás követő bizonytalanságban a közösségi fejlesztők úgy látták, hogy az Oracle-lel való együttműködés kilátásai még rosszabbak lettek, ezrét az OpenOffice-ből leágazva, az addig elkészült programkódra építve egy saját változatot kezdtek el fejleszteni. Az új változat nevének a LibreOffice választották. Érdekes megfigyelni, hogy pont a szabadszoftveres licenc tette lehetővé, hogy az OpenOffice-ra épülve egy másik szoftverprojekt el tudjon indulni. Tovább bonyolította a helyzetet, hogy 2010 novemberében az LibreOffice egyik legjelentősebb támogatóját, a Novellt is felvásárolták. Az eredeti OpenOffice.org programot az Oracle csökkenő támogatása mellett átnevezték Apache OpenOffice.orgra. A 2012-es állás szerint a LibreOffice tűnik a két variáns közül életképesebbnek. De jelenleg nem eldönthető kérdés, hogy tartósan fenntartható két magas színvonalú, szabad szoftveres irodai programcsomag fejlesztése. Érdekes azonban megfigyelni, hogy az üzleti életben gyakran bekövetkező jelentős változások ellenére is folyamatos a szabad szoftver fejlesztése.

A LibreOffice programcsalád számos magyar vonatkozású funkcióval rendelkezik. Magyar fejlesztés a helyesírás-ellenőrző keretrendszer és a numbertext munkalap-függvény is. A magyar helyesírás-ellenőrző több felmérés szerint is jobb teljesítményt nyújt a Microsoft Office beépített helyesírás-ellenőrzőjénél. Mivel a helyesírás-ellenőrző is szabad szoftver, más szoftverekbe is belekerült, így például a Firefox böngésző is azt használja. A numbertext munkalap-függvénnyel számokat tudunk különböző nyelven szöveges formátummá alakítani. Ehhez hasonló funkciójú munkalap-függvény az Excelben nem létezik.

Az ODF és az OOXML formátumok között az átjárás még nem tökéletesen megoldott és a LibreOffice és Microsoft Office sem rendelkezik teljesen azonos funkcionalitással, ezért a Microsoft Office estén jelenleg is beszélhetünk bizonyos mértékű lekötésről.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.

Az Microsoft Office termékcsaládnak más konkurense is akad. A különböző on-line használható irodai programok közül a Google docs szolgáltatása a legnépszerűbb. A szolgáltatást biztosító szoftver ugyan nem szabad szoftver, de a webes szolgáltatás térítésmentesen igénybe lehet venni, és például a dokumentumok egyidejűleg történő közös szerkesztése rendkívül kényelmes. Webes szolgáltatások esetén különösen fontos, hogy mielőtt a felhőbe költöztetnék az adatainkat, vizsgáljuk meg, hogy ez mekkora lekötéssel jár.

Semmelweis Egyetem
Cím: 1085. Budapest, Üllői út 26.
Telefon: +36 (1) 459-1500
E-mail: hirek@semmelweis-univ.hu
Honlap: http://semmelweis-egyetem.hu

MAGYARORSZÁG MEGÚJUL
A projektek az Európai Unió támogatásával valósulnak meg.