



MINISTERUL EDUCAȚIEI ȘI CERCETĂRII

ANDREI FLOREA

SILVIU-EUGEN SĂCUIU



# INFORMATIKA ÉS IKT VII. osztály



EDITURA DIDACTICĂ ȘI PEDAGOGICĂ S.A.





MINISTERUL EDUCAȚIEI ȘI CERCETĂRII

Andrei Florea

Silviu-Eugen Săcuiu

# INFORMATIKA ÉS IKT VII. osztály



EDITURA DIDACTICĂ ȘI PEDAGOGICĂ S.A.

**ACEST MANUAL A FOST FOLOSIT DE:**

Anul	Numele elevului care a primit manualul	Clasa	Școala	Anul școlar	Starea manualului*	
					la primire	la returnare
1.						
2.						
3.						
4.						

\* Starea manualului se va înscrie folosind termenii: nou, bun, îngrijit, nesatisfăcător, deteriorat.  
Cadrele didactice vor controla dacă numele elevului este scris corect.  
Elevii nu trebuie să facă niciun fel de însemnări pe manual.

**Descrierea CIP a Bibliotecii Naționale a României****ANDREI FLOREA****Informatică și Tic : manual pentru clasa a VII-a /Andrei Florea**

Silviu-Eugen Săcuiu. - București : Editura Didactică și Pedagogică, 2019

ISBN 978-606-31-1027-6

I. Săcuiu, Silviu-Eugen

© **E.D.P. 2019.** Toate drepturile asupra acestei ediții sunt rezervate Editurii Didactice și Pedagogice S.A., București. Orice preluare, parțială sau integrală, a textului sau a materialului grafic din această lucrare se face numai cu acordul scris al editurii.

© **Andrei Florea, Silviu-Eugen Săcuiu**

**Referenți:** conf. dr. Sanda Monica Tătărâm – Facultatea de Matematică și Informatică, Universitatea din București  
prof. gr. I Mihaela Anca – Colegiul Național „I.L. Caragiale”

**EDITURA DIDACTICĂ ȘI PEDAGOGICĂ S.A.**

Str. Spiru Haret nr. 12, sector 1, cod 010176, București

Tel.: 021.315.38.20

Tel./fax: 021.312.28.85

e-mail: office@edituradp.ro

web: www.edituradp.ro

Librăria E.D.P.: str. Gen. Berthelot nr. 28-30, sector 1

**Redactori:** Sorin Casapu, Valentina Jercea, Oana Nistor**Tehnoredactor:** Doina Țibea**Coperta:** Alin Casapu**Traducere în lb. maghiară:** Babos László, Budai Edit, Budai István, Erdőközi Enikő, Simon Ilona, Sleghel Izabella, Török Adél, Tóth Zsuzsánna**Coordonatorul traducerii:** Vass Csilla

Comenzile pentru această lucrare se primesc:

– prin poștă, pe adresa editurii

– prin e-mail: comenzi@edituradp.ro;

comercial@edituradp.ro

– prin telefon/fax: 021.315.73.98

# TARTALOMJEGYZÉK

## 1. FEJEZET - SZÖVEGSZERKESZTŐK

1.1. Lecke – A Notepad és a Wordpad szövegszerkesztők.....	10
1.2. Lecke - Microsoft Word – A dokumentum kezelése.....	14
1.3. Lecke - A dokumentumoldal esztétikája .....	17
1.4. Lecke - Szerkesztési műveletek.....	20
1.4.1. - Szerkesztési műveletek egy dokumentumban: kijelölés, másolás, átmozgatás, törlés .....	20
1.4.2. - Szövegformázási műveletek .....	21
1.4.3. - A <i>Font</i> csoportban található gombok szerepe.....	23
1.5. Lecke - Képek formázása .....	27
1.6. Lecke - Táblázat formázása.....	31

## 2. FEJEZET - AUDIO, ILLETVE AUDIO-VIDEO ALKALMAZÁSOK

2.1. Lecke - Hangállományok szerkesztése .....	40
2.1.1. Hangállományokat szerkesztő alkalmazások felhasználói felülete.....	40
2.1.2. Műveletek hangállományok szerkesztéséhez. Hangok rögzítése és lejátszása .....	40
2.2. Lecke - Windows Movie Maker. Bevezetés.....	42
2.2.1. Audio-video szerkesztő alkalmazások felhasználói felülete.....	42
2.2.2. Audio-video szerkesztő alkalmazások kezelésére használt műveletek.....	43
2.3. Lecke - Videoállomány szerkesztése.....	45
2.4. Lecke - Hangsáv és szövegelemek beillesztése.....	47

## 3. FEJEZET KOLLABORATÍV ALKALMAZÁSOK

3.1. Lecke - Kollaboratív alkalmazások jellemzői. Megnyitás / Bejelentkezés .....	54
3.2. Lecke - Kollaboratív alkalmazások lehetőségei és felhasználói felülete.....	58
3.3. Lecke - Munkaeszközök: dokumentumok, bemutatók .....	60
3.4. Lecke - Megengedett műveletek: közzététel, tartalom módosítása .....	62

## 4. FEJEZET - PROGRAMOZÁSI NYELV

4.1. Lecke - Ismétlés: grafikai blokkokkal ábrázolt algoritmusok.....	66
4.2. Lecke - A fejlesztői környezet lehetőségei: szerkesztés, futtatás, hibakeresés.....	79
4.3. Lecke - Operátorok .....	82
4.4. Lecke - Egy program szerkezete.....	84
4.4.1. A nyelv szókészlete.....	84
4.4.2. Numerikus adatok .....	85
4.4.3. Adatok beolvasása és kiírása.....	85
4.4.4. Lineáris szerkezet programozási nyelvbe való átültetésére vonatkozó utasítások.....	87
4.5. Lecke – Elágazási szerkezet implementálása programozási nyelvben .....	89
4.6. Lecke - Ismétlő szerkezet implementálása programozási nyelvben .....	92
4.6.1. A <b>while</b> ismétlő utasítás .....	92
4.6.2. A <b>for</b> ismétlő utasítás .....	94
4.6.3. A <b>do ... while</b> ismétlő utasítás .....	97

<b>HELYES VÁLASZOK.....</b>	<b>100</b>
-----------------------------	------------



# A tankönyv bemutatása

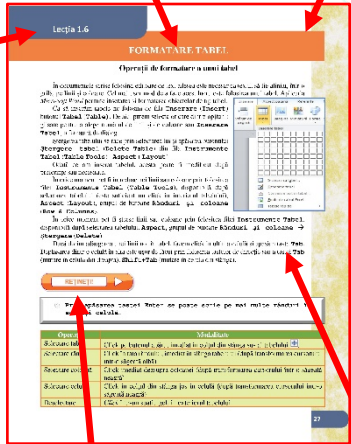


A fejezetet bemutató oldal

A lecke címe

Egy lecke oldalai

A lecke sorszáma



Figyelem!

A lecke tartalma

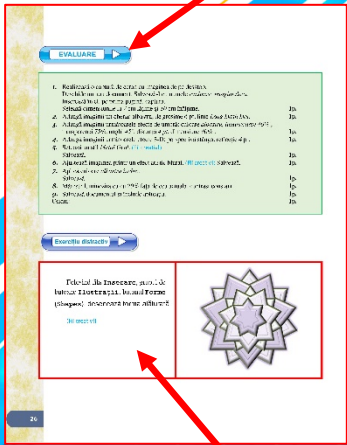
Tudtátok, hogy...?

Jegyzed meg!

Felmérés

Gyakorlati alkalmazások

Projekt

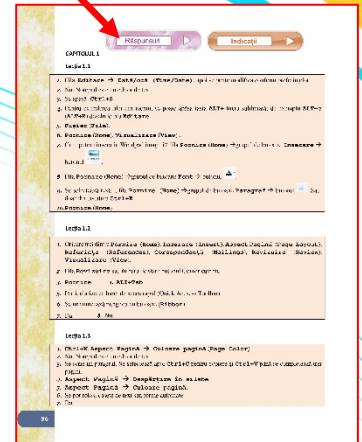
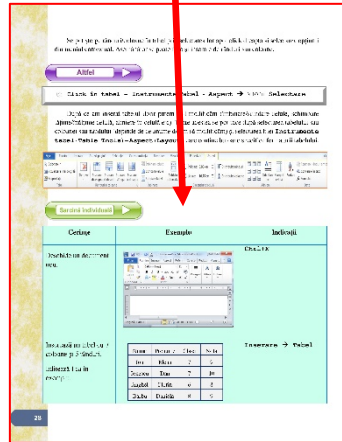
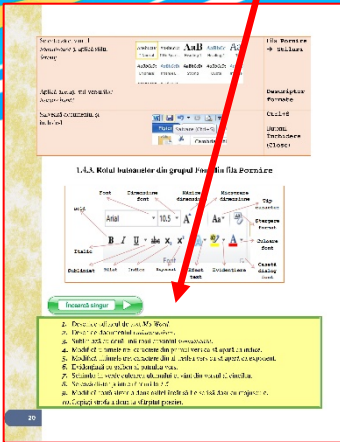


Szórakoztató gyakorlat

Próbálkozz önállóan!

Egyéni feladat

Válaszok és útmutatások



**A következő szimbólumok könnyebbé teszik a tankönyv átolvasását:**

Tudtátok, hogy...?

JEGYEZZÜK MEG!

Próbáld magad!

Gyakorlatok

FIGYELEM!

Másképp

Feladatok

PORTFÓLIÓ

Szórakoztató gyakorlat

Egyéni feladat

FELMÉRŐ

Emlékezzünk vissza!

Útbaigazítások

Válaszok

### Digitális változat



Erre a jelre kattintva nagyíthatod a képet.



Erre a jelre kattintva filmet nézhetsz.



Ha erre kattintsz, elindul egy gyakorlat.



# ÁLTALÁNOS KOMPETENCIÁK

1. Az infokommunikációs technológiák felelős és hatékony felhasználása
2. Elemi feladatok megoldása intuitív információfeldolgozási módszerekkel
3. Társadalmi, művelődési és személyi témájú kreatív kisprojektek kidolgozása, az információk megbízhatóságára és a szerzői jogokra való figyelemmel

## 1. FEJEZET SZÖVEGSZERKESZTÉS

### Sajátos kompetenciák

- 1.1. Dokumentumok szerkesztése célirányos alkalmazások segítségével
  - egy szövegszerkesztő alkalmazás felhasználói felületének tanulmányozása és a fontosabb műveleti lehetőségek beazonosítása
  - sajátos műveletek alkalmazása egy dokumentum szerkesztésekor
  - célirányos eszközök használata egy dokumentum formázására során
- 3.1. Gyakorlati hasznosságú dokumentumok előállítása a tanult alkalmazásokkal
  - dokumentum szerkesztése adott modell alapján, a megadott szabályok betartásával
  - típusdokumentumok szerkesztése: levél, névjegy, diploma, üdvözlőlap stb.



## **2. FEJEZET**

### **HANGFELDOLGOZÓ ALKALMAZÁSOK. AUDIO-VIDEO SZERKESZTŐK**

#### **Sajátos kompetenciák**

- 1.2. Tematikus dokumentáció létrehozása audio, illetve audio-video alkalmazások segítségével
  - interjúkészítés adott témára a tanulók körében, hangrögzítéssel, illetve audio-video formátumú adatrögzítéssel
  - tematikus klip készítése adott állományból, hangszerkesztő, illetve audio-video szerkesztő célirányos műveleteinek felhasználásával
  - audio, illetve audio-video anyag feldolgozása célirányos programeszközök felhasználásával
- 3.2. Audio, illetve audio-video szoftvertermékek létrehozása adott téma bemutatására, a tanult alkalmazások felhasználásával
  - audio-video montázs létrehozása adott forgatókönyv alapján
  - videoklip létrehozása zenei háttérrel, egy iskolai rendezvény bemutatására

## **3. FEJEZET**

### **KOLLABORATÍV ALKALMAZÁSOK**

#### **Sajátos kompetenciák**

- 1.3. Kollaboratív alkalmazások használata digitális termékek csapatmunkában való előállítására
  - kollektív szerző (csoportok, osztály) által írt mese kollaboratív alkalmazásokkal, wiki oldalak vagy blog közös szerkesztése iskola szinten, tantárgyi témakörökben
  - a szerzők hozzájárulásának megjelenítése a „szerkesztési előzmények” eszközzel, önértékelés és egymás közötti kiértékelés céljából
  - helyi azonosítójelek megjelenítése (az iskola vagy a helység képei, közérdekű vagy turisztikai információk stb.) földrajzi információs rendszerekben (GIS)
  - hackerek és crackerek ismertetőjeleinek beazonosítása a lehetséges támadások elleni védekezés céljából, online kollaboratív környezetben
- 3.1. A mindennapi életben használt dokumentumok szerkesztése a tanult alkalmazásokkal
  - Plakát, szórólap stb. előállítása, csapatmunkában, kollaboratív alkalmazásokkal
- 3.2. Audio-video anyagok előállítása adott témára, célirányos alkalmazásokkal
  - audio, illetve audio-video állományok befoglalása (integrálása) más termékekbe: prezentációk, kollaboratív alkalmazások stb.

## 4. FEJEZET PROGRAMOZÁSI NYELV

### Sajátos kompetenciák

- 1.4. Egy programozási környezet felhasználása algoritmusok implementálására
  - egy programozási környezet interfész elemeinek megismerése, a környezet eszközei által biztosított lehetőségek beazonosításának céljából
  - az applikációfejlesztés mozzanatainak alkalmazása egy adott programozási környezetben: forráskód szerkesztése, fordítás (compile), futtatás, tesztelés és hibakeresés az interaktív, grafikus blokkokat használó környezetek lehetőségeihez hasonló módon
  - didaktikai célú robotprogramozás virtuális környezetben, a virtuális robot szenzorai által beolvasott adatok megjelenítése és felhasználása (mint például ultrahang szenzor akadályok beméréséhez, színérzékelő, nyomásérzékelő, mikrofon, infravörös szenzor, giroszkóp szenzor, iránytű stb.)
- 2.1. Egyszerű feladat kijelentésének megoldási algoritmushoz vezető elemzése
  - bemeneti és kimeneti adatok beazonosítása valamely tanult tárgy vagy személyes hobby körébe tartozó ismert feladat esetén
  - a megoldáshoz szükséges műveletek beazonosítása a feladat szerkezeti elemzése alapján
  - a feladat megoldására alkalmas algoritmus beazonosítása
- 2.2. Feladatok megoldására alkalmas elemi algoritmusok építése lineáris utasítások, elágazások, ismétlődő utasítások kombinálásával
  - numerikus adatok feldolgozására alkalmas egyszerű algoritmusok építése grafikus blokkok felhasználásával (például szám számjegyeinek összege, legnagyobb/legkisebb számjegye stb.)
  - algoritmusok építése grafikus blokkok felhasználásával elemi oszthatósági feladatok megoldására (például egy szám osztóinak meghatározása, legnagyobb közös osztó, prímszám beazonosítása)
  - vásárlási költségvetés hatékony kezelésén vagy más egyszerű számításokon alapuló algoritmus előállítása (például számlálás, összegzés, szorzatszámolás)
- 3.3. Algoritmusok implementálása (átültetése) programozási környezetbe, alkalmazás jellegű feladatok kreatív megoldása érdekében
  - forráskód készítése programozási nyelv utasításainak alkalmazásával, mintaalgoritmus alapján
  - létező forráskód futtatása sorról-sorra, a program által használt változók értékeinek követése, annak érdekében, hogy kiderüljenek a tervezési és implementálási hibák
  - egyszerű interdiszciplináris alkalmazások megvalósítása (például: egyenesvonalú, egyenletes mozgású autó sebességének kiszámítása, négyzet, háromszög, trapéz területe és kerülete)
  - a virtuális robot vezérlésére alkalmas forráskód kidolgozása azon parancsok sorozatának megadásával, amelyek elvezetik a robotot egy adott pálya mentén





# **1. FEJEZET**

## **SZÖVEGSZERKESZTÉS**



## A NOTEPAD ÉS WORDPAD SZÖVEGSZERKESZTŐK

Ebben a fejezetben a szövegszerkesztőket szeretnénk bemutatni. Velük gyakran találkozunk, telefonon és táblagépen egyaránt, akkor, amikor valamit csak le szeretnénk jegyezni, de akkor is, amikor valami bonyolultabb szöveget szeretnénk megszerkeszteni.

A következőkben néhány olyan ötletet mutatunk be, amelyek segítenek a szakdolgozatok, képeslapok, poszterek, diplomák, levelek és táblázatok szerkesztésében.

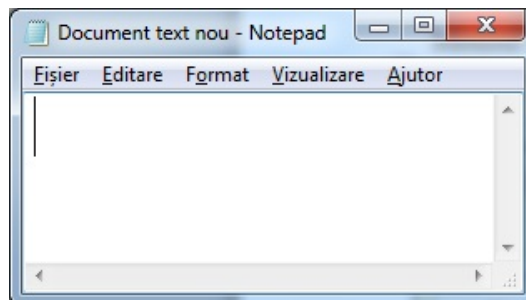
Nem lesz nehéz, csak figyeljete és könnyen fog menni!

Sok sikert a szövegszerkesztéshez!

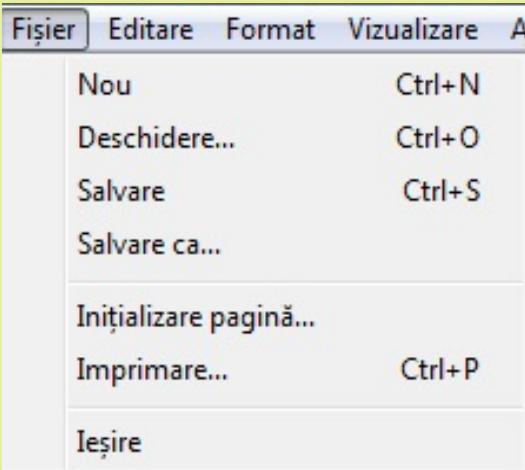
A **szövegszerkesztő** egy olyan program, amely lehetővé teszi a számítógép használatát a szöveg beviteléhez és formázásához. A szöveg formázása során szövegrészeket tudunk kicserélni, törölni, másolni. Hasonlóképpen a szöveg arculatát is tudjuk alakítani: szín, nagyság, tájolás, igazítás.

Szövegszerkesztésre az operációs rendszer speciális alkalmazásait tudjuk használni, vagy más programokat: Notepad, Wordpad, Libre Office, Open Office-Writer, Google Docs.

A **Notepad** (Jegyzetömb) az egyik legegyszerűbb szövegszerkesztő, amely a Microsoft Windows operációs rendszerének része. Kevés szövegszerkesztési lehetőséget tartalmaz. A Notepad szövegszerkesztővel készített állományok kiterjesztése: *txt*.



A menüszalagnak a következő opciói vannak:

Menü	
<b>Fájl (File, Fișier)</b>	
	
	<b>Új (New, Nou)</b> → új dokumentumot hoz létre
	<b>Megnyitás (Open, Deschidere)</b> → megnyit egy létező dokumentumot
	<b>Mentés (Save, Salvare)</b> → lementi a dokumentumban végzett változtatásokat
	<b>Mentés másként (Save As, Salvare ca)</b> → lementi a dokumentumot egy új névvel esetleg egy új helyre, más kiterjesztéssel.
	<b>Oldalbeállítás (Page Setup, Inițializare pagină)</b> → rögzíti a lapbeállításokat (tájolás, margók, fejléc, lábléc)
	<b>Nyomtatás (Print, Imprimare)</b> → megnyitja a nyomtatáshoz szükséges ablakot
	<b>Kilépés (Exit, Ieșire)</b> → bezárja az alkalmazást

## Szerkesztés (Edit, Editare)

Editare	Format	Vizualizare	Ajutor
Anulare			Ctrl+Z
Decupare			Ctrl+X
Copiere			Ctrl+C
Lipire			Ctrl+V
Ștergere			Del
Găsire...			Ctrl+F
Următorul găsit			F3
Înlocuire...			Ctrl+H
Salt la...			Ctrl+G
Selectare totală			Ctrl+A
Dată/Oră			F5

**Visszavonás (Undo, Anulare)** → visszavonja a legutóbb kiadott parancsot.

**Kivágás (Cut, Decupare)** → kivágja a kijelölt szövegrészt a vágólapra

**Másolás (Copy, Copiere)** → kimásolja a vágólapra a kijelölt szöveget

**Beillesztés (Paste, Lipire)** → beilleszti a kivágott vagy kimásolt szöveget

**Törlés (Delete, Ștergere)** → törli a kijelölt szöveget

**Keresés (Find, Găsire)** → megnyitja a keresőablakot

**Csere (Replace, Înlocuire)** → kicserél egy szöveget egy másikra

**Ugrás (Go to, Salt la)** → a kurzort a kívánt sor elejére viszi

**Az összes kijelölése (Select All, Selectare totală)** → kijelöli a teljes szöveget

**Dátum/idő (Time/Date, Dată/oră)** → hozzáadja az aktuális időt, dátumot a szöveghez

## Formátum (Format)

Format	Vizualizare	Ajutor
Încadrare cuvânt		
Font...		

**Sortörés (WordWrap, Încadrare cuvânt)** → úgy igazítja a szöveget, hogy beilleszkedjen a margók közé (azokat a szavakat, amelyek a margón kívül esnek, új sorba teszi).

**Font** → megnyitja a karakterformázásokat tartalmazó ablakot (stílus, nagyság)

Tudtátok, hogy...?



- ⊗ A menü minden opciójára használható a menü jobb oldalán feltüntetett billentyűkombináció?
- ⊗ Csak a billentyűzet segítségével is navigálhatunk a menüelemek között (**ALT** + aláhúzott betű)?
- ⊗ Ha a menü egy opciója szürke, akkor az a menü nem használható?

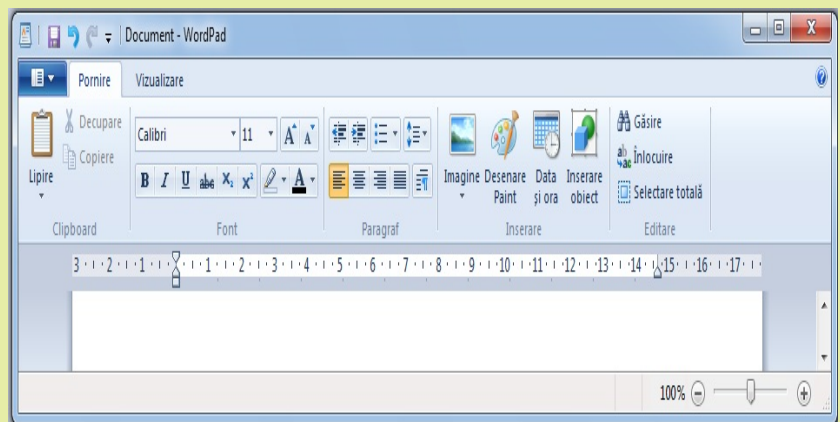
1. Nyisd meg a *Notepad* szövegszerkesztőt.
2. Írd be a mellékelt szöveget<sup>1</sup>.
3. Mentsd le a szöveget *eminescu* néven.
4. Mi az állomány kiterjesztése?
5. Módosítsd fekvő tájolására a dokumentumot (*landscape*).
6. Állítsd a margók szélességét *1 inch*-re.
7. Helyettesítsd mindenhol a szövegben az *or* betűcsoportot *aa* betűcsoporttal.
8. Vond vissza az előbbi változtatást.
9. Az egész szövegben alkalmazd a következő fontbeállításokat: *Times New Roman, Bold, Italic, 14*
10. Mentsd le, majd zárd be a dokumentumot és az applikációt.

*Somnoroase păsărele  
Pe la cuiburi se adună,  
Se ascund în rămuirele –  
Noapte bună!  
Doar izvoarele suspină,  
Pe când codrul negru tace;  
Dorm și florile-n grădină –  
Dormi în pace!*

## Tudtátok, hogy...?

- ⊗ Egy lehetőség Notepad használata helyett a Notepad++?  
(ez lehetővé teszi C++, HTML stb. forráskódok szerkesztését)

**WordPad** a Notepad szövegszerkesztőnél bővebb lehetőségeket nyújtó alkalmazás, amely a *Microsoft Windows* operációs rendszerének a része. A dokumentumokat *.rtf (Rich Text Format)* kiterjesztéssel menti le. A Notepad szövegszerkesztőhöz képest, itt a szöveget lehet igazítani, ki lehet emelni, színezní, aláhúzni, áthúzni. Lehet felső és alsó indexbe írni. Lehet listákat készíteni és kép- illetve rajzobjektumokat beszúrni.



<sup>1</sup> Részlet Mihai Eminescu: *Somnoroase păsărele* című verséből



1. Nyisd meg a *WordPad* szövegszerkesztőt.
2. Írd be a mellékelt szöveget<sup>2</sup>.
3. Mentsd le a szöveget *eminescu* néven, *.rtf* kiterjesztéssel.
4. A lap méretét módosítsd *A4*-es méretűre.
5. Az oldalak margóit állítsd *1 inch*-re.
6. Az egész szöveg betűi legyenek: *Times New Roman, Bold, Italic, 14*
7. Változtasd meg az *or* szövegrészek formázását az egész szövegben a következőképpen: piros, aláhúzott, sárgával kiemelt.
8. Szűrd be az aktuális dátumot és órát a dokumentum végére.
9. Mentsd le, majd zárd be a dokumentumot és az alkalmazást.

*Somnoroase păsărele  
Pe la cuiburi se adună,  
Se ascund în rămurele –  
Noapte bună!  
Doar izvoarele suspină,  
Pe când codrul negru tace;  
Dorm și florile-n grădină –  
Dormi în pace!*

## FELMÉRŐ

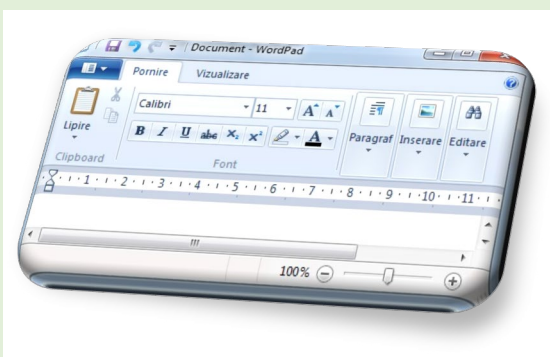
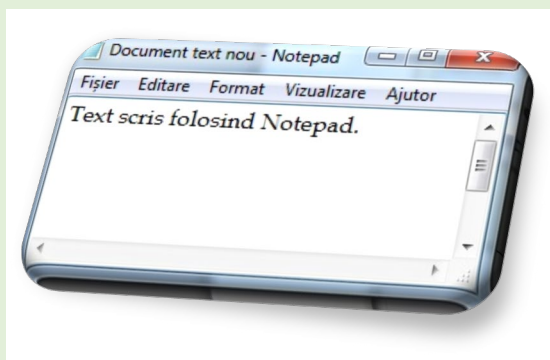
Válaszolj a következő kérdésekre:

Pontszám

- Példa:**  
Hogy tudjuk eltüntetni a többszörös szóközöket egy szövegből, úgy, hogy minden két szó között pontosan egy szóköz legyen?
1. Hogy szerkeszthető a dátum a Notepad szövegszerkesztőben? 1p.
  2. Használható a Notepad, mint 3D-s animációs alkalmazás? Indokold meg! 1p.
  3. Hogy menthetünk le a Notepad szövegszerkesztőben egy dokumentumot, csak billentyűzet segítségével? 1p.
  4. Miért van aláhúzva minden menüpontban egy betű? 0,5p.
  5. Melyik menüpont tartalmazza a lapra vonatkozó beállításokat? 1p.
  6. Mit tartalmaz a WordPad menüszalag? 0,5p.
  7. Hogy tudunk beilleszteni képet Wordpad-ben? 1p.
  8. Milyen gombra kattintunk, hogy más színnel írjunk? 1p.
  9. Hogy igazítjuk középre a szöveget? 1p.
  10. Melyik menüpont tartalmazza a szövegkeresési lehetőséget? 1p.
- Hivatalból 1p.

## Útbaigazítások

**Editare (Edit) → Înlocuire (Replace)**  
A két szóköz karaktert helyettesítjük egy szóköz karakterrel, addig ismételjük az eljárás ameddig nem történik több helyettesítés



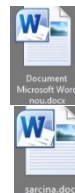
<sup>2</sup> Részlet Mihai Eminescu: *Somnoroase păsărele* című verséből

## MICROSOFT WORD – DOKUMENTUMKEZELÉS

A formázási lehetőségeket tekintve sokkal fejlettebb szövegszerkesztő a **Microsoft Word**. Ezzel, azon kívül, hogy lehetőségünk van elvégezni azokat a formázásokat is, amelyeket a többi szövegszerkesztő kínál, szerkeszthetünk jelentéseket, rajzokat, képeket, táblázatokat (az utóbbiakat csak bizonyos mértékig, hiszen sokkal jobb alkalmazások léteznek erre a célra).

A Word alkalmazással készült dokumentumok felismerhetők a mellékelt ikonról:

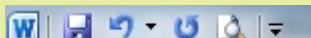
A Microsoft Word-dal készült dokumentumok nevét a **.docx** kiterjesztés követi.



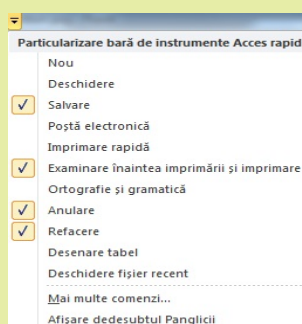
Tudtátok, hogy...?

- ⊗ Az alkalmazás teljes neve **Microsoft Office Word**?
- ⊗ A formátuma **.doc** volt a Word 2003-as verzióig, **.docx** formátumú a Word 2007-től kezdődően?
- ⊗ **doc** az angol *document* szóból származik?

Az operációs rendszer verziójától függően a *Microsoft Word* szövegszerkesztő megnyitásához, a Start → Microsoft Office → Microsoft Word lépéseket követjük, vagy begépeljük az operációs rendszer keresőjébe a Microsoft Word szöveget. A legtöbb esetben a *Office* csomag telepítése után az asztalon megtalálható a *Microsoft Word* szövegszerkesztő parancsikonja.



A gyorselérési eszköztár (Quick Access Toolbar).  
Megváltoztatható a felhasználó igényei szerint.



Fisier Pornire Inserare Aspect pagină Referințe Corespondență Revizuire Vizualizare



Szalag (Ribbon) – csoportokban rendezve tartalmazza a parancsokat. Ez szalagfülenként változik.

Pagină: 7 din 27 Cuvinte: 4.224 Română 120%

Állapotsor – az aktuális állapotról tartalmaz információkat.

## Dokumentumkezelő<sup>3</sup> műveletek

Művelet	Módszer
Új dokumentum létrehozása	<b>Fájl (File, Fișier) → Új (New, Nou) →</b> Kiválasztunk egy sablon típust → <b>Létrehozás (Create, Creare)</b> Használva a <b>Ctrl+N</b> (New) billentyűkombinációt
Létező dokumentum megnyitása	<b>Fájl → Megnyitás (Open, Deschidere)</b> Használva a <b>Ctrl+O</b> (Open) billentyűkombinációt Majd kiválasztjuk a kért dokumentumot
Több dokumentum megnyitása	Megismételjük az előbbi lépéseket minden dokumentumra <b>Nézet (View, Vizualizare) → Ablakváltás (Switch Windows, Comutare ferestre)</b>
Dokumentum mentése	<b>Fájl (File, Fișier) → Mentés (Save, Salvare)</b> Csak a billentyűzetet használva <b>Ctrl+S</b> (Save) Ha a dokumentum még nem volt lementve használhatjuk a <b>Fájl (File, Fișier) → Mentés másként (Save as, Salvare ca)</b> lépéseket, és adunk egy nevet.
Dokumentum mentése más néven, más helyre vagy más kiterjesztéssel Másolatkészítés az aktuális dokumentumról	<b>Fájl (File, Fișier) → Mentés másként</b> Átnevezzük a dokumentumot, majd (választás szerint) <b>Mentés másként</b> és kiválasztjuk a kívánt formátumot (.txt, .rtf, .pdf, .dotx)
Dokumentum bezárása	<b>Fájl (File, Fișier) → Bezárás (Close, Închidere)</b>
Alkalmazás bezárása	Az alkalmazásablak jobb felső sarkában található  gomb vagy <b>(Alt+F4)</b> vagy <b>Fájl (File, Fișier) → Kilépés (Exit, Ieșire)</b> vagy a bal felső sarokban található  gomb

### Gyakorlatok

1. Nyisd meg a *Microsoft Word* szövegszerkesztőt.
2. Mentsd a dokumentumot 7 néven, *docx* kiterjesztéssel.
3. Mentsd a dokumentumot *sapte* néven, *pdf* kiterjesztéssel.
4. A gyorselérési eszköztárhoz add hozzá a *Megnyitás (Open, Deschidere)* parancsot.
5. Rejtsd el (minimalizáld) a szalagot. Jegyezd meg milyen billentyűkombinációt használtál!
6. Hozd elő a szalagot.
7. Állítsd be a nagyítást 115%-ra
8. Állítsd be a mértékegységet *centiméterre*.
9. Mentsd le a dokumentumot, zárd be a dokumentumot anélkül, hogy az alkalmazást bezárnád.

<sup>3</sup> V. osztályban tanultátok az információk szervezését állományokban és mappákban.






**FELMÉRŐ**

Válaszolj a következő kérdésekre:

**Példa:**

Egy mód, hogy a mértékegységet *inch*-ről *centiméter*-re változtassuk a **Fájl (File, Fișier) → Beállítások (Options, Optiuni) → Speciális (Advanced, Avansat) → Megjelenítés (Display, Ecran)** lépésekkel történik.

1. Sorolj fel három szalagfület Microsoft Word menüszalagról. 1p.
2. Egy módszer a szavak megszámlálására a dokumentumban: ..... 1p.
3. A dokumentum mentése a következő menüpontban található: ..... 1p.
4. Hogy tudunk váltani két megnyitott dokumentum között? **(Légy kreatív!)** 2p.
5. Milyen szerepe van a  gombnak? 1p.
6. Mi történik **Ctrl+F1** billentyűkombináció használatakor? 1p.
7. Készíthető egy új dokumentum csak billentyűzet segítségével?  Igen  Nem 1p.
8. Az egyetlen módszer a **Zoom** megváltoztatására a **Nézet (View, Vizualizare)** menüpontban található.

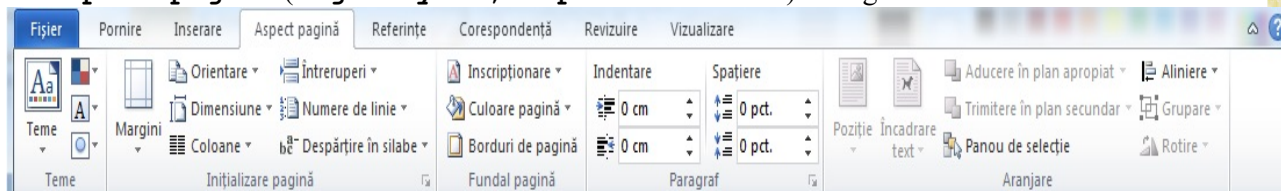
Igen  Nem 1p.

Hivatalból 1p.

## A dokumentumoldal esztétikája

### Szövegszerkesztési szabályok és a nyomtatott oldal esztétikája


Az **Aspect pagină** (Page Layout, Lap elrendezése) szalagfűlön:



a következő lehetőségeink vannak:

Művelet	Módszer
A dokumentum lapbeállításai: margók, tájolás, lapméret	Az Oldalbeállítás (Page Layout, Inițializare pagină) csoport
Elválasztás (Hyphenation, Despărțire în silabe)	
Sorok számozása (Line Numbers, Numere de linii)	
Vízjel (Inscripționare, Watermark)	A Oldalháttér (Page Background, Fundal pagină) csoport
Oldalszín (Page Color, Culoare pagină)	
Oldalszegélyek (Page Borders, Borduri de pagină)	

A **Nézet** (View, Vizualizare) szalagfűlön a következő lehetőségek vannak:

Művelet	Módszer
A dokumentum különböző nézetben való megtekintése <b>Nyomtatási elrendezés</b> (Print layout, Aspect pagină imprimată), <b>Olvasómód</b> (Read mode, Citire în ecran complet), <b>Webes elrendezés</b> (Web layout, Aspect pagină Web), <b>Vázlat</b> (Outline, Schiță), <b>Piszkozat</b> (Draft, Ciornă)	A <b>Nézetek</b> (Document Views, Vizualizări documente) csoport vagy a jobb alsó sarokban található gombok 
A nézet méretének a változtatása ( <b>Zoom</b> )	A <b>Nagyítás (Zoom)</b> csoport
Több oldal megtekintése a képernyőn	A <b>Nagyítás (Zoom)</b> csoport
Az ablak vízszintes felosztása és a dokumentum két részének megtekintése egy időben.	Az <b>Ablak</b> (Window, Fereastră) → <b>Felosztás</b> (Split, Scindare) lépésekkel
Az ablak függőleges elosztása és két dokumentum egyidőben való megtekintése	A <b>Ablak</b> (Window, Fereastră) → <b>Párhuzamos nézet</b> (View Side by Side, Vizualizare alăturate) lépésekkel
A vízszintes/függőleges vonalzó láthatóvá tétele	Az <b>Megjelenítés</b> (Show, Afișare) csoport
A nyomtatási kép megtekintése	<b>Fájl</b> (File, Fișier) → <b>Nyomtatás</b> (Print, Imprimare)

## JEGYEZZÜK MEG!

- ⊗ A nézet méretének változtatása (**Zoom**) nincs hatással a lapméretre.
- ⊗ A nyomtatási kép megtekintése hasznos az utolsó javítások elvégzéséhez.  
**Fájl (File, Fișier) → Nyomtatás (Print, Imprimare)**
- ⊗ Két egymás mellé megnyitott dokumentumot össze tudunk hasonlítani Microsoft Word-ben, hogy különbségeket találjunk, a **Párhuzamos nézet (View Side by Side, Vizualizare alăturată)** segítségével.

## Tudtátok, hogy...?

- ⊗ Ha egy oldalnak háttérszínt állítunk be, akkor a többi oldal háttérszíne is ugyanaz lesz.

## Gyakorlatok

1. Nyisd meg a *Microsoft Word* szövegszerkesztőt.
2. Mentsd le a dokumentumot *pagina* néven, *.docx* kiterjesztéssel.
3. Állítsd a nézetet **Nyomtatási elrendezés (Print layout, Aspect pagină imprimată)** módra .
4. Állítsd be a nagyítást 115%-ra.
5. Tedd láthatóvá a vonalzót.
6. Állíts minden margót 2,5 cm-re, a kötésmargót 2 cm-re, a lap méretét *A4-re*, tájolását *álló-ra*.
7. Alkalmazd 2pt vastagságú, piros, folytonos vonal keretet az oldalra.
8. Alkalmazd a *pagina* vízjelt az oldalra, *Arial, 44, szürke, átlós, áttetsző* beállításokkal.
9. Állítsd a paragrafus előtti távolságot *6 pt*-ra, a paragrafus utánit *0 pt*-ra .
10. Mentsd és zárd be a dokumentumot anélkül, hogy bezárnád az alkalmazást.



### Mielőtt kinyomtatónál egy dokumentumot:

- nézd meg és ellenőrizd az oldal kinézetét;
- ellenőrizd a **Minden látszik (Show/Hide, Afişare/Ascundere)** a **Kezdőlap (Home, Pornire)** szalagfűlből a **Bekezdés (Paragraph, Paragraf)**, csoportból a szavak közti szóközök számát (két egymásmelletti szó között egyetlen szóköz kell legyen);
- ellenőrizd a beírt és üres sorok arányát;
- ellenőrizd a képek méretét és tisztaságát.



Válaszolj a következő kérdésekre:

#### Példa:

Alkalmazd egy beállítást, amely a lapot A4-es méretűre teszi.

Válasz: **Elrendezés (Layout, Aspect pagină)** szalagfűl → **Méret (Size, Dimensiune)** → **A4**

1. Nyiss meg egy új dokumentumot. Állítsd be tetszőleges színűre az oldal háttérszínét. 1p.
  2. Alkalmazd képekből álló margót. 1p.
  3. Írj egy tetszőleges bekezdést. Másold le úgy, hogy három legyen belőle az oldalon. 1p.
  4. Alkalmazd automatikus szótagolást. 1p.
  5. Alkalmazd háttérszín az első oldalra. 1p.
  6. Állíts az 1-es és 3-as oldalakra **azonos**, a 2-esre pedig **más** háttérszín. (Légy kreatív!) 1p.
  7. Megtekinthető egyszerre több oldal egy dokumentumból?  Igen  Nem 1p.
  8. Miért hasznos a vonalzó, amikor szerkesztünk egy oldalt? 1p.
  9. A kötésmargó beállításának célja ..... 1p.
- Hivatalból: 1p.

## SZERKESZTÉSI MŰVELETEK

### 1.4.1. Szerkesztési műveletek egy dokumentumban: kijelölés, másolás, mozgatás, törlés

Egy karakter vagy több karakterből álló szöveg kijelölésére használt egyszerű módszerek:

- egér használatával, lenyomott bal egérgombbal végig húzzuk az egeret,
- a **Shift +** nyilak segítségével.

De használhatunk más gyorsabb módszereket is:

Művelet	Módszer
Szó kijelölése	Dupla kattintás a szóra
Szöveg kijelölése	<b>Ctrl+kattintás</b> a szövegre
Bekezdés kijelölése	Hármas kattintás az bekezdése
Teljes szöveg kijelölése	<b>Ctrl+A (All)</b>
Kijelölés megszüntetése	kattintás egy üres helyre
Szöveg másolása (előtte kijelöljük a másolandó szöveget!)	<ul style="list-style-type: none"> <li>o kattintás a <b>Kezdőlap (Home, Pornire)</b> szalagfülből a <i>Vágólap (Clipboard)</i> csoportból a ikonra</li> <li>o <b>Ctrl+C</b></li> <li>o Jobb kattintás → <i>Másolás (Copy, Copiere)</i></li> </ul>
Szöveg mozgatása (kivágása) (előtte kijelöljük az átmozgató szöveget!)	<ul style="list-style-type: none"> <li>o kattintás a <b>Kezdőlap (Home, Pornire)</b> szalagfülből a <i>Vágólap (Clipboard)</i> csoportból a <i>Kivágás (Cut, Decupare)</i> ikonra</li> <li>o <b>Ctrl+X</b></li> <li>o Jobb kattintás → <i>Kivágás (Cut, Decupare)</i></li> </ul>
Szöveg beillesztése a dokumentumba (beillesztés)	<ul style="list-style-type: none"> <li>o kattintás a <b>Kezdőlap (Home, Pornire)</b> szalagfülből a <i>Vágólap (Clipboard)</i> csoportból a <i>Lipire (Paste)</i> ikonra</li> <li>o <b>Ctrl+V</b></li> <li>o Jobb kattintás → <i>Beillesztés (Paste, Lipire)</i></li> </ul>
Szöveg keresése	<ul style="list-style-type: none"> <li>o kattintás a <b>Kezdőlap (Home, Pornire)</b> szalagfülből a <i>Szerkesztés (Editare, Edit)</i> csoportból a <i>Keresés (Find, Găsire)</i> ikonra</li> <li>o <b>Ctrl+F</b></li> </ul>
Szövegcsere	<ul style="list-style-type: none"> <li>o kattintás a <b>Kezdőlap (Home, Pornire)</b> szalagfülből a <i>Szerkesztés (Editare, Edit)</i> csoportból a <i>Csere (Replace, Înlocuire)</i> ikonra</li> <li>o <b>Ctrl+H</b></li> </ul>
Szöveg törlése (előtte kijelöljük a törölni kívánt szöveget!)	<ul style="list-style-type: none"> <li>o A <b>Delete</b> billentyű (törli a karaktereket jobbra)</li> <li>o A <b>Backspace</b> billentyű (törli a karaktereket balra)</li> </ul>
Parancs visszavonása	<ul style="list-style-type: none"> <li>o a  parancsgomb a gyorselérési eszköztárból</li> <li>o a <b>Ctrl+Z</b> billentyűkombináció használatával</li> </ul>
Parancs újravégrehajtása	<ul style="list-style-type: none"> <li>o a  parancsgomb a gyorselérési eszköztárból</li> <li>o a <b>Ctrl+Y</b> billentyűkombináció használatával</li> </ul>

JEGYEZZÜK MEG!



⊗ Nem minden parancs vonható vissza!

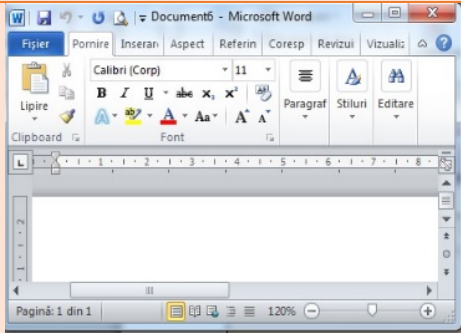

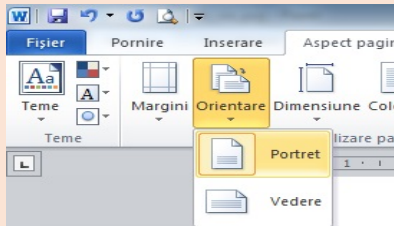
- dokumentum mentése
- dokumentum nyomtatása

*Magyarázzátok meg miért!*

## 1.4.2. Szövegformázási műveletek

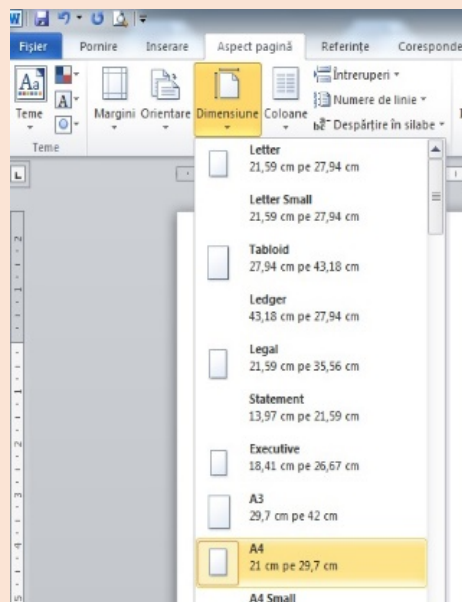
Gyakorlatok



	Példa	Útbaigazítás
Nyisd meg a <i>Microsoft Word</i> szövegszerkesztőt és készíts egy új dokumentumot.		<b>Ctrl+N</b>
Mentsd le a dokumentumot <i>eminescu.docx</i> néven.		<b>Fájl (File, Fișier) → Mentés másként (Save as, Salvare ca)</b>
Állítsd át az oldalt <i>fekvő</i> tájolására.		<b>Oldalbeállítás (Page Setup, Aspect pagină) → Tájolás (Orientation, Orientare) – Fekvő (Landscape, Vedere)</b>

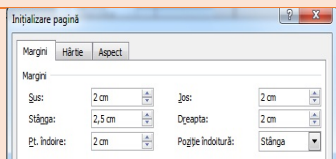


Állítsd a lapméretet *A4*-re.



Oldalbeállítás (Page Setup, Aspect pagină) → Méret (Size, Dimensiune)

Állítsd be a margókat: felső 2 cm, alsó 2 cm, bal 2,5 cm, jobb 2 cm, kötésmargó 2 cm.



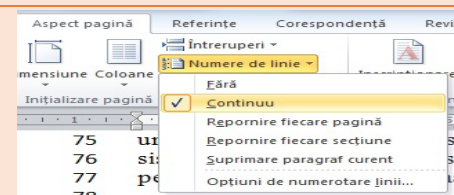
Elrendezés (Layout, Aspect pagină) → Oldalbeállítás (Page Setup, Inițializare pagină)

Írd be a mellékelt szöveget a dokumentumba<sup>4</sup>.

Somnoroase păsărele  
Pe la cuiburi se adună,  
Se ascund în rămurele –  
Noapte bună!  
Doar izvoarele suspină,  
Pe când codrul negru tace,  
Dorm și florile-n grădină –  
Dormi în pace!

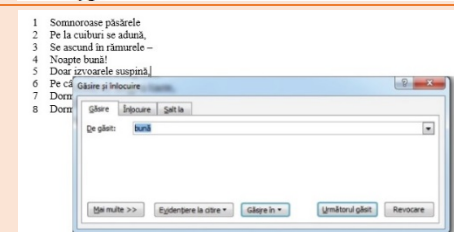
Trece lebăda pe ape  
Între trestii să se culce –  
Fie-ți îngerii aproape,  
Somnul dulce!  
Peste-a nopții feerie  
Se ridică mândra lună,  
Totu-i vis și armonie –  
Noapte bună!

Állítsd be a sorok számozását.



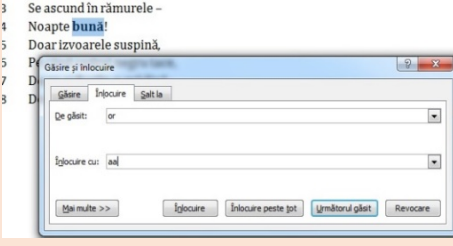
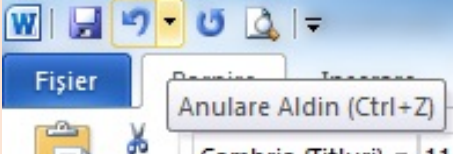

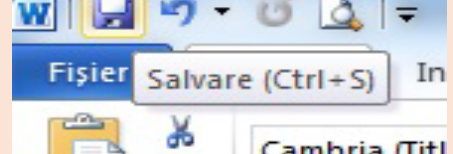
Elrendezés (Layout, Aspect pagină) szalagfűl → Sorok számozása (Line Numbers, Numere de linii)

Keress meg *bună* szót és formázd meg félkövér betűstílust, zöld színt, két vonallal aláhúzást használva.



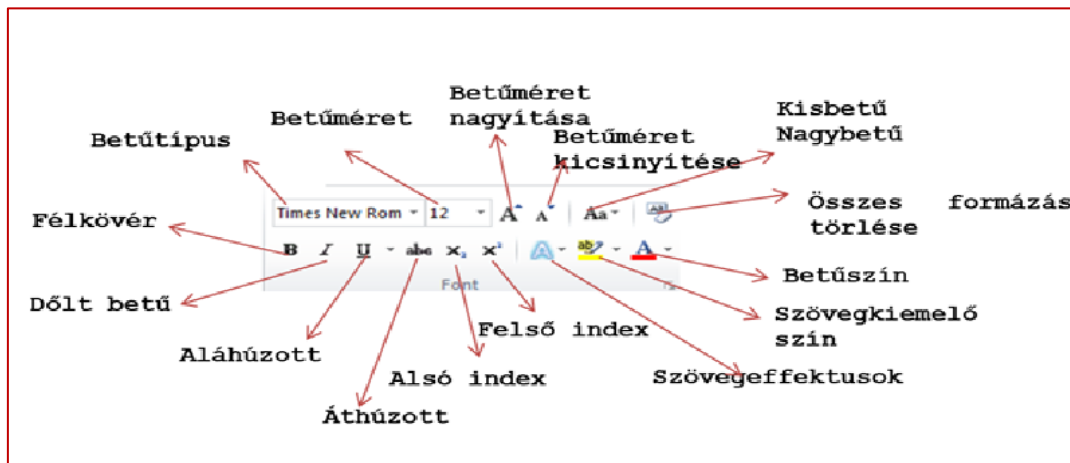
Ctrl+F → Következő (Găsește următorul) gomb → Ctrl+B → Aláhúzott (Underline, Subliniat) → Betűszín (Font color, Culoare font)

<sup>4</sup> Részlet Mihai Eminescu: *Somnoroase păsărele* című verséből

<p>Cseréld ki mindenhol a szövegben az <i>or</i> betűcsoportot az <i>aa</i> betűcsoportra.</p>		<p><b>Ctrl+H</b> → <b>Az összes cseréje (Replace all, Înlocuire peste tot)</b> parancs</p>
<p>Vond vissza az előbbi változtatást.</p>		<p><b>Ctrl+Z</b></p>
<p>Jelöld ki az első sort és másold az utolsó után.</p>		<p><b>Ctrl+C</b> <b>Ctrl+V</b></p>
<p>Jelöld ki a <i>Dorm și florile...</i> kezdetű sort és mozgasd a <i>Între trestii...</i> kezdetű sor után.</p>		<p><b>Ctrl+X</b> <b>Ctrl+V</b></p>
<p>Jelöld ki <i>Somnoroase</i> szöveget és alkalmazd a <i>Strong</i> stílust.</p>		<p><b>Kezdőlap (Home, Pornire) → Stílusok (Styles, Stiluri)</b></p>
<p>Alkalmazd ugyanazt a stílust <i>Noaptea bună!</i> verssorra.</p>		<p><b>Formátummásoló (Format Painter, Descriptor formate)</b></p>
<p>Mentsd és zárd be a dokumentumot.</p>		<p><b>Ctrl+S</b> <b>Bezár (Close, Închidere)</b> gomb</p>

### 1.4.3. A Kezdőlap (Home, Pornire) szalagfü

#### Betűtípus (Font) csoportjának parancsai



Próbáld magad!



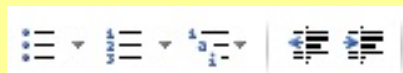
1. Nyisd meg az *MS Word* szövegszerkesztőt.
2. Nyisd meg az *eminescu.docx* dokumentumot.
3. Húzd alá piros dupla vonallal a *somnoroase* szót.
4. Formázd alsó indexként az első sor utolsó három karakterét.
5. Formázd felső indexként az első sor utolsó három karakterét.
6. Emeld ki sárgával a negyedik sort.
7. Az ötödik verssor utolsó szavának színét változtasd zöldre.
8. Állítsd *l,5*-re a sorközt.
9. Alakítsd a teljes második versszakot csupa nagybetűssé.
10. Másold a második szakaszt a vers végére.



## FELMÉRŐ

1. Hozz létre egy új dokumentumot.  
Mentsd el **ertekeles\_font.docx** néven.  
Nyisd meg azt a dokumentumot, amelybe előzőleg a *Somnoroase pășărele* c. verset írtad. **1p.**
  2. Másold a szöveget az új dokumentumba 5-ször egymás alá. Zárd be azt a dokumentumot, amiből másoltál. **1p.**
  3. A *Noapte bună!* sor minden előfordulását igazítsd középre. Használd a megfelelő eszközt. **1p.**
  4. Állíts be tetszőleges háttérszínt az utolsó sorban. (Légy kreatív!) **1p.**
  5. Igazíts a jobb margóhoz minden sort az ötödiktől a tizenkettedik sorig **1p.**
  6. Cseréld ki a versben a karaktereket Arial, 18 pontosra, válassz egy effektust. (Légy kreatív!)  
Csökkentsd a karakterek méretét felére. **1p.**
  7. Keretezd be tetszőleges módon a második sorban a *cuiburi* szót. **1p.**
  8. Igazítsd a vers címét középre.  
A cím alatt a szerző nevét igazítsd jobbra.  
Illessz be két üres sort. **1p.**
  9. Alkalmazz 3 pt vastag, dupla vonalas zöld keretet a címre.  
Mentsd el és zárd be a dokumentumot. **1p.**
- Hivatalból: **1p.**

## Egyéni feladat



A **Bekezdés (Paragraph, Paragraf)** csoport eszközeit használva hozd létre a mellékelt listát.

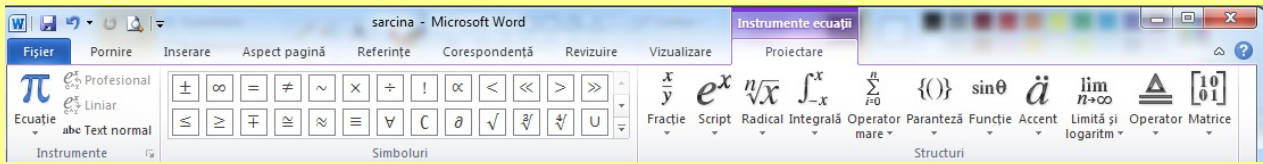
**Betűtípus (Font)** *Times New Roman, 12, Félkövér*

(Légy kreatív!)

Listă:

- Creion
  - ◇ Roșu
  - ◇ Verde
  - ◇ Negru
- Pix
  - Albastru
  - Roșu
- Caiet
  1. A4
  2. A5
- Manual
- Culegere
  - ✓ Matematică
    - Clasa a VI-a
    - Clasa a VII-a
  - ✓ Informatică
- Riglă

Egyéni feladat



A **Beillesztés (Insert, Inserare)** menü **Szimbólumok (Symbols, Simbol)** csoport **Egyenletek (Equation, Ecuatie)** elemének segítségével szerkeszd meg a következő kifejezéseket.

$$\frac{a}{b} + \sqrt{x^{\frac{m}{n}}} + (c^{5^b + [c^s]}) \neq \sqrt{a^2 + b^2} \cdot \frac{\pi^s}{2}$$

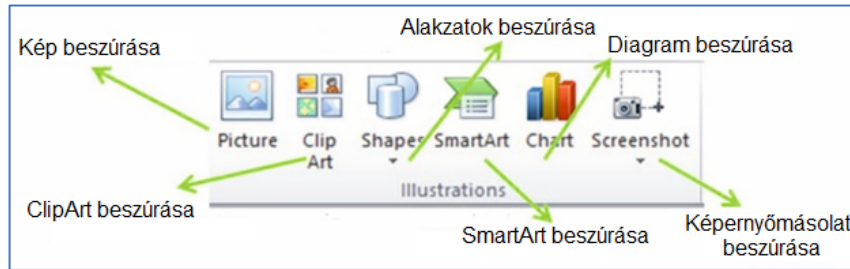
$$E(x, y, \bar{v}) = \frac{[x^{2^{x+1}} \pm \max(x, y)]}{(y^{y+2^{-x}} + \sqrt{|x-y|}) \left[ \frac{x}{x+2 \times y} \right] - |\bar{v}| + y} + f(x, y)$$

$$\text{ahol } f(x, y) = \begin{cases} -x + \frac{y}{2}, & x \leq 0 \\ x + \sqrt{|x + y^{3^y}|}, & \text{a többi esetben} \end{cases}$$

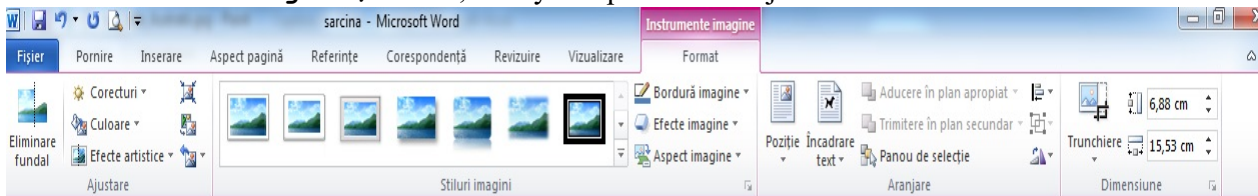
## KÉPEK FORMÁZÁSA

### Egy kép formázásának lépései

Egy dokumentumban nagyon gyakran kísérik a szöveget képek is. A Microsoft Word alkalmazás lehetővé teszi a kép típusú objektumok beillesztését és formázását. Egy kép beillesztésére a **Beillesztés (Insert, Inserare)** menü **Ábrák (Illustrations, Ilustrații)** csoportjának **Kép (Picture, Imagine)** elemét használjuk.



Miután beillesztettük a kívánt képet, módosíthatjuk (méretét, megjelenését, fényerejét) vagy kiegészíthetjük új elemekkel (effektusok, keret). Mindezeket megtehetjük, ha kijelöljük a képet (ráklikkelünk az egérrel) és kiválasztjuk a **Kép-eszközök (Picture Tools, Instrumente imagine)** menüt, amely a képformázás sajátos eszközeit tartalmazza.



Egy képet ki lehet törölni, ha kiválasztjuk és megnyomjuk a **Delete** vagy **Backspace** gombot.







A képet át lehet helyezni a **Másolás/Kivágás (Copy/Cut, Copiere/Decupare)** és **Beillesztés (Paste, Lipire)** gombok segítségével vagy a nyíl billentyűkkel.


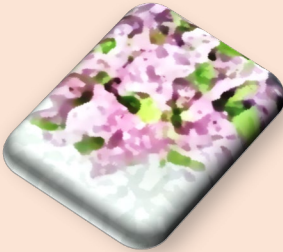
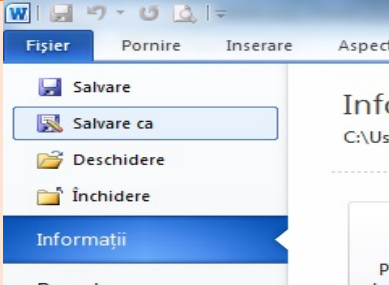

### Gyakorlatok



Követelmény	Példa	Útmutatás
Nyiss meg egy új dokumentumot.		<b>Ctrl+N</b>



<p>Illessz az aktuális dokumentumba egy képet a háttértárolóról.</p>		<p><b>Beillesztés</b> (Insert, Inserare) → <b>Kép</b> (Picture, Imagine)</p>
<p>Állítsd be a kép méreteit: 8 cm magasra és 5 cm szélesre.</p>		<p><b>Képeszközök</b>, (Picture Tools, Instrumente imagine) → <b>Formatum (Format)</b> → <b>Méret csoport</b> (Size, Dimensiune) Figyelj az arányosságra!</p>
<p>Forgasd el a képet 30°-al jobbra.</p>		<p><b>Képeszközök</b>, (Picture Tools, Instrumente imagine) → <b>Formatum (Format)</b> → <b>Forgatás</b> (Rotate, Rotare)</p>
<p>Keretezd be a képet egy 3 pt vastag piros szaggatott vonallal.</p>		<p><b>Képeszközök</b>, (Picture Tools, Instrumente imagine) → <b>Formatum (Format)</b> → <b>Képszegély</b> (Picture Border, Bordură imagine)</p>
<p>Állíts be jobbra-fel vetülő árnyékot a képnek</p>		<p><b>Képeszközök</b>, (Picture Tools, Instrumente imagine) → <b>Formatum (Format)</b> → <b>Képeffektusok</b> (Picture Effects, Efecte imagine) → <b>Árnyék (Shadow, Umbră)</b></p>
<p>Forgasd el a képet perspektivikusan balra.</p>		<p><b>Képeszközök</b>, (Picture Tools, Instrumente imagine) → <b>Formatum (Format)</b> → <b>Képeffektusok</b> (Picture Effects, Efecte imagine) → <b>Térbeli forgatás (3-D Rotation, Rotație 3-D)</b></p>

<p>Változtasd meg tetszőlegesen a kép stílusát.</p>		<p><b>Képeszközök</b> , (Picture Tools, Instrumente imagine) → <b>Formatum (Format)</b> → <b>Képstílusok</b> (Picture Styles, Stiluri imagini)</p>
<p>Alkalmazd egy tetszőleges művészi effektust.</p>		<p><b>Képeszközök</b> , (Picture Tools, Instrumente imagine) → <b>Formatum (Format)</b> → <b>KépEffektusok</b> (Artistic Effects, Efecte artistice)</p>
<p>Mentsd el a dokumentumot .pdf formátumban.</p>		<p><b>Fájl</b> (File, Fișier) → <b>Mentés másként</b> (Save as, Salvare ca) → <b>.pdf</b></p>
<p>Mentsd el és zárd be a dokumentumot.</p>		<p><b>Ctrl+S</b> <b>Bezárás</b> (Close, Închidere) gomb</p>

**JEGYEZZÜK MEG!**



⊗ Ha egy ideig rajta tartod az egérkurzort egy eszközön, a kép az eszköz által javasolt megjelenésű lesz.

1. Készíts egy képernyőmásolatot az asztalról (desktop).  
Hozz létre egy új dokumentumot, mentsd el *kep\_felmero.docx* néven.  
Illeszd be az első oldalra a képernyőképet.  
Állítsd be a kép méreteit: 7 cm széles és 10 cm magas. 1p.
  2. Keretezd be a képet 6 pt vastag, kék hosszú vonal és pont (*Long Dash Dot*) vonallal. 1p.
  3. Állíts be árnyékolást a következő tulajdonságokkal: *kék* szín, 40% fényerő, 75% áttetszőség, 45°-os szög, 4 pt távolság, 90% méret. 1p.
  4. Állíts be 3D effektusokat: bal oldali perspektíva, 4 pt reflexió. 1p.
  5. Válassz egyfémszerű ovális (*Metal Oval*) stílust. **(Légy kreatív!)**  
Mentsd el. 1p.
  6. Állíts be egy elmosódó művészi effektust. **(Légy kreatív!)** Mentsd el. 1p.
  7. Állíts be *sötétkék* színt.  
Mentsd el. 1p.
  8. Növeld meg a fényerőt 20%-kal, a kontraszt maradjon változatlan. 1p.
  9. Mentsd el a dokumentumot és zárd be az alkalmazást. 1p.
- Hivatalból: 1p.

## Szórakoztató gyakorlat

A **Beszúrás** (Insert, Inserare) menü **Ábrák** (Illustrations, Ilustrații) csoportjának **Alakzatok** (Shapes, Forme) eszközét használva készítsd el a mellékelt rajzot. **(Légy kreatív!)**





# TÁBLÁZAT FORMÁZÁSA

## A táblázat formázásának eszközei

Egy szöveg szerkesztésekor gyakran van arra szükség, hogy a szöveget egy rácsszerkezetbe, sorokra és oszlopokra osztva rendezzük. Ennek legegyszerűbb módja a táblázatok használata. A *Microsoft Word* alkalmazás lehetővé teszi a táblázatok beszúrását és formázását.

Egy táblázat beszúrására a **Beszúrás (Insert, Inserare)** menü **Táblázat (Table, Tabel)** gombját/eszközét használjuk. Itt két lehetőségünk van a sorok és oszlopok számának megadására: vagy az egérkurzor mozgatásával választjuk ki, vagy a **Táblázat beszúrása (Insert Table, Inserare Tabel)** párbeszédablakba írjuk be.

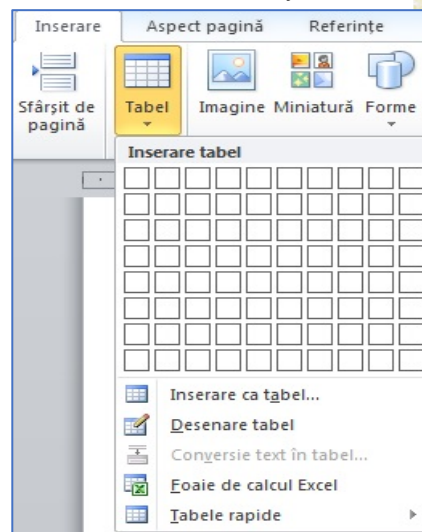
Törléshez ki kell a táblázatot jelölni, majd a **Táblázat Eszközök (Table Tools, Instrumente Tabel)** menü **Megjelenés (Layout, Aspect)** csoportjának **Táblázat törlése (Delete Table, Ștergere tabel)** eszközt választva törlőd.

Ha beillesztettük a táblázatot, lehet szükség szerint módosítani.


Bármikor hozzá lehet adni új sorokat vagy oszlopokat a **Táblázat Eszközök (Table Tools, Instrumente Tabel)** menü **Megjelenés (Layout, Aspect)** csoportjának, **Sorok és oszlopok (Row & Columns, Rânduri și coloane)** eszköze segítségével. A menü akkor válik hozzáférhetővé, ha kijelöljük a táblázatot (elég, ha táblázaton belül letesszük az egérkurzort).

Bármikor törölhetünk sorokat vagy oszlopokat ugyanott a **Törlés (Delete, Ștergere)** eszközzel. Ha a táblázathoz új sort akarunk illeszteni, akkor az utolsó cellába visszük a kurzort és leütjük a **Tab** billentyűt. A cellák közti mozgás legegyszerűbb a nyíl billentyűket használva vagy a **Tab** billentyű (egy cellával jobbra léptet), **Shift+Tab** (egy cellával balra léptet) segítségével.

JEGYEZZÜK MEG!



⊗ Az Enter gomb leütése új sort kezd a cellán belül.

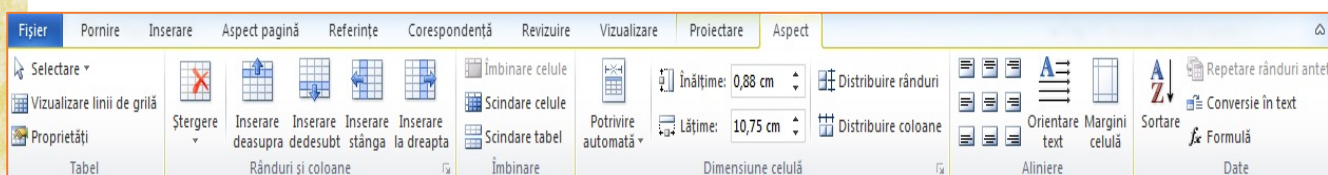
Művelet	Megoldási mód
Táblázat kijelölése	Klikk a táblázat bal felső sarkán megjelenő  jelre.
Sor kijelölése	Klikk közvetlenül a táblázat bal szélénél (ha az egérkurzor fehér jobbra nyíl)
Oszlop kijelölése	Klikk közvetlenül az oszlop fölött (ha az egérkurzor fekete lefele nyíl)
Cella kijelölése	Klikk a cella bal alsó sarkára (ha az egérkurzor fekete ferde nyíl)
Kijelölés megszűnése	Klikk egy üres területre a táblázaton kívül.

Lehet sort/oszlopot törölni úgy is, hogy kijelölöm őket és a jobb klikk segítségével megnyitott helyi menüből választom a törlést. Hasonló módon sorokat vagy oszlopokat be is lehet szűrni.

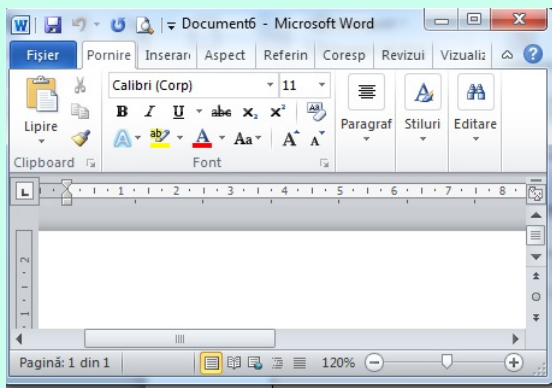
Másképp

⊗ **Klikk a táblázatba → Táblázateszközök (Table Tools, Instrumente Tabel) menü, → Elrendezés (Layout, Aspect) csoport → Kijelölés (Select, Selectare) eszköz.**

Miután beszúrtuk a táblázatot, módosíthatjuk (egyesíthetünk, feloszthatunk cellákat, megváltoztathatjuk a cellák szélességét, magasságát, igazíthatunk a cellán belül stb.). Ehhez ki kell jelölni a táblázatot vagy annak sorait vagy oszlopait (attól függően, hogy mit akarok módosítani) majd kiválasztani a **Táblázateszközök (Table Tools, Instrumente Tabel) menü Elrendezés (Layout, Aspect) csoportjának** megfelelő eszközét.



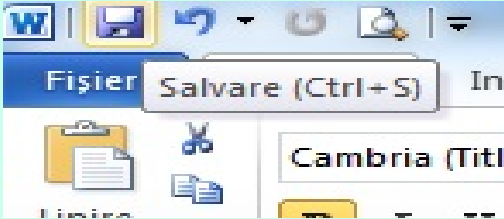
Egyéni feladat



Követelmény	Példa	Utasítás																				
Nyiss meg egy új dokumentumot.		<b>Ctrl+N</b>																				
Szűrj be egy 4 oszlopos és 5 soros táblázatot. Írj bele adatokat a példa szerint.	<table border="1"> <thead> <tr> <th>Név</th> <th>K. név</th> <th>Oszt.</th> <th>Jegy</th> </tr> </thead> <tbody> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> </tbody> </table>	Név	K. név	Oszt.	Jegy	János	Rózsi	7	9	Jancsi	Dani	7	10	Antal	Ferenc	6	8	Béla	Dia	8	9	<b>Beszúrás → Táblázat</b> (Insert, Inserare → Tabel)
Név	K. név	Oszt.	Jegy																			
János	Rózsi	7	9																			
Jancsi	Dani	7	10																			
Antal	Ferenc	6	8																			
Béla	Dia	8	9																			

<p>Módosítsd a sorok magasságát 2 cm-re.</p>	<table border="1"> <thead> <tr> <th>Név</th> <th>K. név</th> <th>Oszt.</th> <th>Jegy</th> </tr> </thead> <tbody> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> </tbody> </table>	Név	K. név	Oszt.	Jegy	János	Rózsi	7	9	Jancsi	Dani	7	10	Antal	Ferenc	6	8	Béla	Dia	8	9	<p><b>Táblázateszközök</b> (Table Tools, Instrumente Tabel) → <b>Elrendezés</b> (Layout, Aspect) → <b>Cellaméret</b> (Cell Size, Dimensiune celulă)</p>				
Név	K. név	Oszt.	Jegy																							
János	Rózsi	7	9																							
Jancsi	Dani	7	10																							
Antal	Ferenc	6	8																							
Béla	Dia	8	9																							
<p>Módosítsd az oszlopok szélességét 1 cm-re.</p>	<table border="1"> <thead> <tr> <th>Név</th> <th>K. név</th> <th>Oszt.</th> <th>Jegy</th> </tr> </thead> <tbody> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> </tbody> </table>	Név	K. név	Oszt.	Jegy	János	Rózsi	7	9	Jancsi	Dani	7	10	Antal	Ferenc	6	8	Béla	Dia	8	9	<p><b>Táblázateszközök</b> (Table Tools, Instrumente Tabel) → <b>Elrendezés</b> (Layout, Aspect) → <b>Cellaméret</b> (Cell Size, Dimensiune celulă)</p>				
Név	K. név	Oszt.	Jegy																							
János	Rózsi	7	9																							
Jancsi	Dani	7	10																							
Antal	Ferenc	6	8																							
Béla	Dia	8	9																							
<p>Formázd át a táblázat fejlécét: <i>középre igazított, 14 pt-os Arial, dőlt, csupa nagybetű</i> legyen</p>	<table border="1"> <thead> <tr> <th><i>NÉV</i></th> <th><i>K. NÉV</i></th> <th><i>OSZT.</i></th> <th><i>JEGY</i></th> </tr> </thead> <tbody> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> </tbody> </table>	<i>NÉV</i>	<i>K. NÉV</i>	<i>OSZT.</i>	<i>JEGY</i>	János	Rózsi	7	9	Jancsi	Dani	7	10	Antal	Ferenc	6	8	Béla	Dia	8	9	<p><b>Kezdőlap</b> (Home, Pornire) → <b>Betűtípus</b> (Font) → <b>Bekezdés</b> (Paragraph, Paragraf)</p>				
<i>NÉV</i>	<i>K. NÉV</i>	<i>OSZT.</i>	<i>JEGY</i>																							
János	Rózsi	7	9																							
Jancsi	Dani	7	10																							
Antal	Ferenc	6	8																							
Béla	Dia	8	9																							
<p>Módosítsd a táblázat vonalait (szín, vastagság)</p>	<table border="1"> <thead> <tr> <th><i>NÉV</i></th> <th><i>K. NÉV</i></th> <th><i>OSZT.</i></th> <th><i>JEGY</i></th> </tr> </thead> <tbody> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> </tbody> </table>	<i>NÉV</i>	<i>K. NÉV</i>	<i>OSZT.</i>	<i>JEGY</i>	János	Rózsi	7	9	Jancsi	Dani	7	10	Antal	Ferenc	6	8	Béla	Dia	8	9	<p>A táblázat kijelölése, jobb egér klikk → <b>Szegély és Mintázat</b> (Borders and Shading, Borduri și umbrire)</p>				
<i>NÉV</i>	<i>K. NÉV</i>	<i>OSZT.</i>	<i>JEGY</i>																							
János	Rózsi	7	9																							
Jancsi	Dani	7	10																							
Antal	Ferenc	6	8																							
Béla	Dia	8	9																							
<p>Adj hozzá egy új sort a táblázathoz.</p>	<table border="1"> <thead> <tr> <th><i>NÉV</i></th> <th><i>K. NÉV</i></th> <th><i>OSZT.</i></th> <th><i>JEGY</i></th> </tr> </thead> <tbody> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	<i>NÉV</i>	<i>K. NÉV</i>	<i>OSZT.</i>	<i>JEGY</i>	János	Rózsi	7	9	Jancsi	Dani	7	10	Antal	Ferenc	6	8	Béla	Dia	8	9					<p>Az utolsó cella kiválasztása, majd <b>Tab</b> billentyű.</p>
<i>NÉV</i>	<i>K. NÉV</i>	<i>OSZT.</i>	<i>JEGY</i>																							
János	Rózsi	7	9																							
Jancsi	Dani	7	10																							
Antal	Ferenc	6	8																							
Béla	Dia	8	9																							



<p>Egyesítsd az utolsó sor első két celláját.</p>	<table border="1"> <thead> <tr> <th>NÉV</th> <th>K. NÉV</th> <th>OSZT.</th> <th>JEGY</th> </tr> </thead> <tbody> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> <tr> <td colspan="2"></td> <td></td> <td></td> </tr> </tbody> </table>	NÉV	K. NÉV	OSZT.	JEGY	János	Rózsi	7	9	Jancsi	Dani	7	10	Antal	Ferenc	6	8	Béla	Dia	8	9					<p>A cellák kijelölése,  → <b>Táblázateszközök</b>  (Table Tools, Instrumente Tabel)  → <b>Elrendezés</b>  (Layout, Aspect)  → <b>Cellák egyesítése</b>  (Merge, Îmbinare)</p>						
NÉV	K. NÉV	OSZT.	JEGY																													
János	Rózsi	7	9																													
Jancsi	Dani	7	10																													
Antal	Ferenc	6	8																													
Béla	Dia	8	9																													
<p>Színezd ezt a cellát sárgára.</p>	<table border="1"> <thead> <tr> <th>NÉV</th> <th>K. NÉV</th> <th>OSZT.</th> <th>JEGY</th> </tr> </thead> <tbody> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> <tr> <td colspan="2"></td> <td></td> <td></td> </tr> </tbody> </table>	NÉV	K. NÉV	OSZT.	JEGY	János	Rózsi	7	9	Jancsi	Dani	7	10	Antal	Ferenc	6	8	Béla	Dia	8	9					<p>A cella kijelölése,  <b>Kezdőlap</b>  (Home, Pornire)  → <b>Bekezdés</b> csoport  (Paragraph, Paragraf)</p>						
NÉV	K. NÉV	OSZT.	JEGY																													
János	Rózsi	7	9																													
Jancsi	Dani	7	10																													
Antal	Ferenc	6	8																													
Béla	Dia	8	9																													
<p>Illessz be egy <b>A</b> nevű új oszlopot a <b>K. NÉV</b> és az <b>OSZT.</b> közé</p>	<table border="1"> <thead> <tr> <th>NÉV</th> <th>K. NÉV</th> <th>A</th> <th>OSZT.</th> <th>JEGY</th> </tr> </thead> <tbody> <tr> <td>János</td> <td>Rózsi</td> <td></td> <td>7</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td></td> <td>7</td> <td>10</td> </tr> <tr> <td>Antal</td> <td>Ferenc</td> <td></td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td></td> <td>8</td> <td>9</td> </tr> <tr> <td colspan="2"></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	NÉV	K. NÉV	A	OSZT.	JEGY	János	Rózsi		7	9	Jancsi	Dani		7	10	Antal	Ferenc		6	8	Béla	Dia		8	9						<p>Jelöld ki az oszlopot (K.NÉV vagy OSZT.) jobb egér klikk, <b>Beszúrás</b>  (Insert, Inserare)</p>
NÉV	K. NÉV	A	OSZT.	JEGY																												
János	Rózsi		7	9																												
Jancsi	Dani		7	10																												
Antal	Ferenc		6	8																												
Béla	Dia		8	9																												
<p>Rendezd a táblázat adatait név szerint.</p>	<table border="1"> <thead> <tr> <th>NÉV</th> <th>K. NÉV</th> <th>A</th> <th>OSZT.</th> <th>JEGY</th> </tr> </thead> <tbody> <tr> <td colspan="2"></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Antal</td> <td>Ferenc</td> <td></td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td></td> <td>8</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td></td> <td>7</td> <td>10</td> </tr> <tr> <td>János</td> <td>Rózsi</td> <td></td> <td>7</td> <td>9</td> </tr> </tbody> </table>	NÉV	K. NÉV	A	OSZT.	JEGY						Antal	Ferenc		6	8	Béla	Dia		8	9	Jancsi	Dani		7	10	János	Rózsi		7	9	<p>Kijelöld a táblázatot,  → <b>Elrendezés</b>  (Layout, Aspect)  → <b>Rendezés</b>  (Sort, Sortare)</p>
NÉV	K. NÉV	A	OSZT.	JEGY																												
Antal	Ferenc		6	8																												
Béla	Dia		8	9																												
Jancsi	Dani		7	10																												
János	Rózsi		7	9																												

<p>Töröld az első (adat) sort.</p>	<table border="1"> <thead> <tr> <th>NÉV</th> <th>K. NÉV</th> <th>OSZT.</th> <th>JEGY</th> </tr> </thead> <tbody> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> </tbody> </table>	NÉV	K. NÉV	OSZT.	JEGY	Antal	Ferenc	6	8	Béla	Dia	8	9	Jancsi	Dani	7	10	János	Rózsi	7	9	<p>Kijelölöd a sort, jobb egér klikk, <b>Törlés</b> (Delete Row, Ștergere)</p>
NÉV	K. NÉV	OSZT.	JEGY																			
Antal	Ferenc	6	8																			
Béla	Dia	8	9																			
Jancsi	Dani	7	10																			
János	Rózsi	7	9																			
<p>Igazítsd a táblázatban a szöveget vízszintesen és függőlegesen középre.</p>	<table border="1"> <thead> <tr> <th>NÉV</th> <th>K. NÉV</th> <th>OSZT.</th> <th>JEGY</th> </tr> </thead> <tbody> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> </tbody> </table>	NÉV	K. NÉV	OSZT.	JEGY	Antal	Ferenc	6	8	Béla	Dia	8	9	Jancsi	Dani	7	10	János	Rózsi	7	9	<p>Kijelölsz egy sort, jobb egér klikk, <b>Cellaigazítás</b> (Cell Alignment, Aliniere celulă)</p>
NÉV	K. NÉV	OSZT.	JEGY																			
Antal	Ferenc	6	8																			
Béla	Dia	8	9																			
Jancsi	Dani	7	10																			
János	Rózsi	7	9																			
<p>A táblázat első sorának állíts be más színű és vastagabb keretet.</p>	<table border="1"> <thead> <tr> <th>NÉV</th> <th>K. NÉV</th> <th>OSZT.</th> <th>JEGY</th> </tr> </thead> <tbody> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> </tbody> </table>	NÉV	K. NÉV	OSZT.	JEGY	Antal	Ferenc	6	8	Béla	Dia	8	9	Jancsi	Dani	7	10	János	Rózsi	7	9	<p>Kijelölöd a sort, jobb egér klikk <b>Szegély és Mintázat</b> (Borders and Shading, Borduri și umbrire)</p>
NÉV	K. NÉV	OSZT.	JEGY																			
Antal	Ferenc	6	8																			
Béla	Dia	8	9																			
Jancsi	Dani	7	10																			
János	Rózsi	7	9																			
<p>Változtasd meg a táblázat stílusát.</p>	<table border="1"> <thead> <tr> <th>NÉV</th> <th>K. NÉV</th> <th>OSZT.</th> <th>JEGY</th> </tr> </thead> <tbody> <tr> <td>Antal</td> <td>Ferenc</td> <td>6</td> <td>8</td> </tr> <tr> <td>Béla</td> <td>Dia</td> <td>8</td> <td>9</td> </tr> <tr> <td>Jancsi</td> <td>Dani</td> <td>7</td> <td>10</td> </tr> <tr> <td>János</td> <td>Rózsi</td> <td>7</td> <td>9</td> </tr> </tbody> </table>	NÉV	K. NÉV	OSZT.	JEGY	Antal	Ferenc	6	8	Béla	Dia	8	9	Jancsi	Dani	7	10	János	Rózsi	7	9	<p>Kijelölöd a táblázatot <b>Tervezés</b> (Design, Proiectare) → <b>Táblastílusok</b> (Table Styles, Stiluri tabel)</p>
NÉV	K. NÉV	OSZT.	JEGY																			
Antal	Ferenc	6	8																			
Béla	Dia	8	9																			
Jancsi	Dani	7	10																			
János	Rózsi	7	9																			
<p>Mentsd el és zárd be a dokumentumot.</p>		<p><b>Ctrl+S</b> <b>Bezárás</b> eszköz (Close, Închidere)</p>																				

1. Nyiss meg egy új dokumentumot. Mentsd el *tabla\_felmero.docx* néven.  
Szűrj be egy 5 soros és 6 oszlopos táblázatot. **1p.**  
Az első sor tartalma legyen: *Név, Keresztnév, Helység, Életkor, Osztály, Átlag.*
  2. Töltsd fel a táblázat sorait az első sornak megfelelő adatokkal. Mentsd el. **1p.**
  3. Állíts be minden oszlopra azonos szélességet.  
Állíts be minden sorra azonos magasságot. **1p.**
  4. Igazítsd a beírt szöveget a táblázatban.  
A vonalak színét állítsd kékre.  
Módosítsd a táblázat keretét duplavonalasra. **1p.**  
Mentsd el.
  5. Rendezd a táblázatban levő adatokat *Átlag* szerint csökkenő sorrendbe. **1p.**
  6. Melyik eszközt kell kiválasztani, hogy a táblázat keretei láthatók legyenek?  
  **Borduri** (rácsvonalak) **1p.**  
  **Linii de grilă** (szegélyek)
  7. Állítsd be, hogy a szöveg távolsága a cellahatároktól *0,2* cm legyen. **1p.**
  8. Szűrj be egy új sort a második és a harmadik sor közé.  
Minden cellát színezz más-más színűre. **1p.**
  9. Válassz *2. jelölőszínes (Accent 2)* stílust a táblázatnak **(Légy kreatív!)** **1p.**  
Mentsd el a dokumentumot és zárd be az alkalmazást.
- Hivatalból: **1p.**

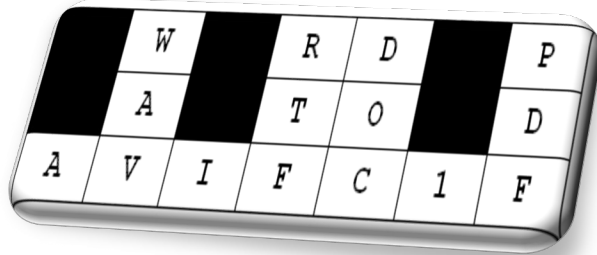




A **Beszúrás** (Insert, Inserare) menü eszközeit használva készítsd el a mellékelt ábrát. (Légy kreatív!)



Keress alapértelmezett állománykiterjesztéseket.



### Gyakorlatok



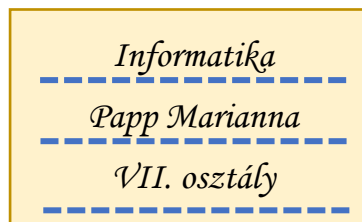
1. Szerkessz egy tematikus beszámolót egy az osztályban kapott témára!
2. Minden szülő számára hozz létre egy üdvözlő kártyát, mentsd el .pdf formátumba majd küldd el őket e-mailben a szülőknek.
3. Hozz létre egy cv .docx állományt, amely a saját CV-det tartalmazza.
4. Hozz létre egy plakátot Microsoft Word-ben egy adott témával kapcsolatban.
5. Szerkessz meg egy névjegykártya mintát .pdf típusú állományban.
6. Hozz létre egy diplomát .pdf formátumban valamelyik osztálytársad számára.
7. Írj egy levelet valamelyik családtagodnak. Ha elkészültél, küldd el a levelet e-mailben.
8. Az eddig tanultakat felhasználva szerkeszd meg a tankönyv következő oldalát.
9. **Portfólió.** Hozz létre egy mappát, amelyben legyen benne az IKT órákon szerkesztett összes dokumentumod.

## Gyakorlati alkalmazás 1

Szerkesszünk *címként* az informatika füzet számára.

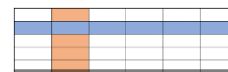
### Lépések:

- A **Beszúrás (Insert, Inserare)** menüben kiválasztunk egy téglalap alakzatot.
- Kitöltjük a **téglalapot** halvány sárgával, bekeretezzük sötét sárgával.
- Megrajzolunk egy szaggatott vonalat, beállítjuk a vastagságát és a színét. Többször egymás alá másoljuk.
- A **Beszúrunk (Insert, Inserare)** egy **Szövegdobozt (TextBox)**. Beírjuk az **Informatika** szöveget.
- A szövegdobozt többször egymás alá másoljuk. Átírjuk a szöveget: **Papp Marianna** illetve **VII. osztály**.
- Csoportosítjuk az elemeket: kijelöljük őket, majd helyi menü - **Csoportok (Group, Grupare)**.



## Gyakorlati alkalmazás 2.

Gyors szerkesztés – Táblázat



Ha be szeretné illeszteni egy táblázatot a dokumentum egy adott pozíciójába, akkor gépelj + és - jeleket, újra + és - , újra + és - - és így tovább, majd üss Entert. *Próbáld ki!*

Példa: +--++--

## Gyakorlati alkalmazás 3.

Lábjegyzet



### Lépések:

1. A kurzort elhelyezzük a szövegben az után a szó után, amihez lábjegyzetet akarunk illeszteni.
2. A **Hivatkozás (References, Referințe)** menüből, **Lábjegyzetek (Footnotes, Note de subsol)** csoportból kiválasztjuk a **Lábjegyzet beszúrása (Insert Footnote, Inserare nota de subsol)** eszközt.
3. A lap aljára beírjuk a megfelelő megjegyzést.

A lábjegyzet<sup>1</sup> a lap alján jelenik meg.  
A lábjegyzet<sup>2</sup> száma vagy szimbóluma megfelel a szövegen belüli hivatkozásnak.

<sup>1</sup> Ez egy lábjegyzet.

<sup>2</sup> Ez egy másik lábjegyzet



**2. FEJEZET**  
**AUDIO**  
**ILLETVE**  
**AUDIO-VIDEO**  
**ALKALMAZÁSOK**



## HANGÁLLOMÁNYOK SZERKESZTÉSE

Feltetted már magadnak a kérdést hogyan kell hangállományokat rögzíteni?

1876-ban, Alexander Graham Bell feltaláló szabadalmaztatta az első olyan szerkezetet, amely képes volt küldeni és fogadni hangokat.

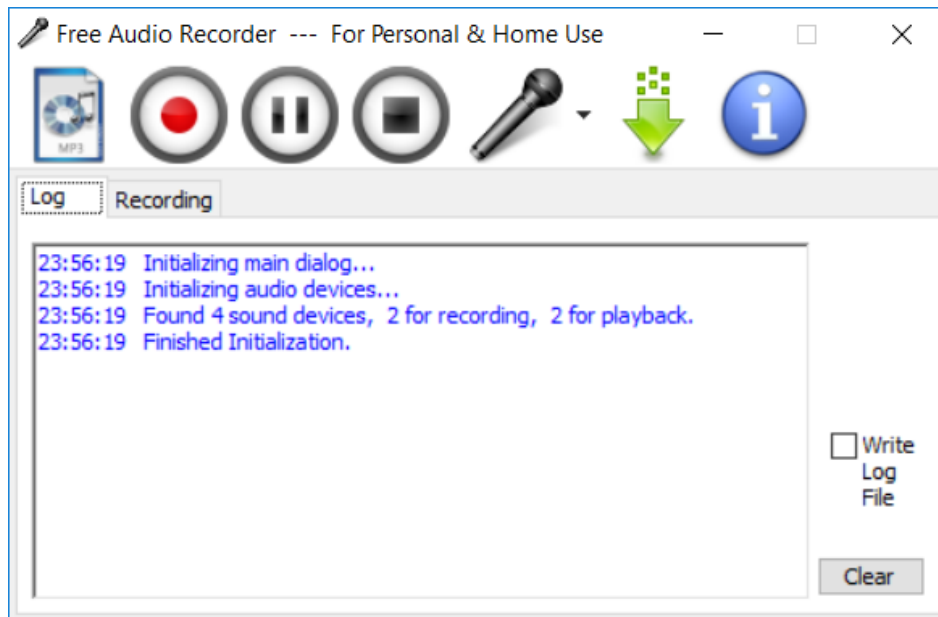
Tudtátok, hogy...?



- ⊗ az első rádióhullámok által kiadott jel közvetítése 1896. március 24-én történt meg és Alexander Popovnak köszönhető?


### 2.1.1. Audioállományokat feldolgozó alkalmazás kezelési felülete

Nagyon sok olyan alkalmazás létezik, amelynek segítségével hangállomány hozható létre. Az egyik legkönnyebben használható ezek közül a „Free Audio Recorder”.



### 2.1.2. Audioállományokat feldolgozó alkalmazás műveletei.


#### Hangok rögzítése és visszajátszása


Egy hanganyag rögzítésének megkezdéséhez szükséges létrehozni az állományt. Ehhez kattelnünk kell a menü második gombjára, „Start recording”  és ki kell választanunk a létrehozás és mentés helyét, valamint az audioállomány nevét.

JEGYEZZÜK MEG!



Az audiorögzítés akkor kezdődik, amikor beírtuk az állomány nevét és elmentettük.

A hangrögzítés befejezéséhez kattintunk a „Stop”  gombra. Amint a „Stop” gombot aktiváltuk, a hangállományt már csak lejátszásra tudjuk megnyitani.

Rögzítés közben, ha szükséges, leállhatunk egy bizonyos időre a „Pause”  gomb lenyomásával. A rögzítés újraindítására megint le kell nyomni a „Pause” gombot.

Egy hangállomány lejátszására a következő alkalmazások közül választhatunk:



AIMP



Windows Media Player

Tudtátok, hogy...?



⊗ a delfinek és a denevérek hanghullámokat használnak, hogy egy mentális „vázlatot” készítsenek a helymeghatározáshoz?

Gyakorlatok



1. Használd a hangrögzítésre alkalmas alkalmazást a saját bemutatkozó állományod elkészítésére. Az állomány maximum 1 perces legyen.
2. A hangrögzítő alkalmazás segítségével hozz létre egy állományt, amelyben Románia himnuszát szavalod.

## BEVEZETÉS A Windows Movie Maker-be

Szeretnétek megtanulni, hogyan kell a vakációban elkészített filmeket megszerkeszteni? Vagy készíteni egy filmet az általatok készített fotókból? Lehet, hogy láttátok a barátaitok által megosztott filmeket a Youtube-on, Facebook-on vagy Whatsapp-on és tetszettek?

Továbbá, megtudhatjátok hogyan készítesek kisfilmeket a rendelkezésetekre álló anyagokból. Operatőrök, rendezők és főképp videoszerkesztők lesztok!

Elsősorban szükségetek lesz az általatok vagy a születek által elkészített fotókra. Ezeket letöltitek a számítógépre, ez alkotja a projektetek kiindulópontját. Jó tudni, hogy képeitek tulajdonképpen, nagyrészt, JPEG formátumban vannak, filmjeitek pedig, amelyeket mobiltelefonnal készíttetek vagy letöltöttetek a Youtube-ról, MP4 formátumban.

Ezekkel a “forrásokkal” indulunk útnak!

Sokan közületek készíttetek már kisfilmeket mobiltelefonnal vagy más készülékkel. Azok a videofájlok nagyon nehezen szerkeszthetők erre alkalmas alkalmazások nélkül. Ezek közül az egyik a Windows Movie Maker. Könnyen használható alkalmazás, mivel nagyon intuitív.

Tudtátok, hogy...?

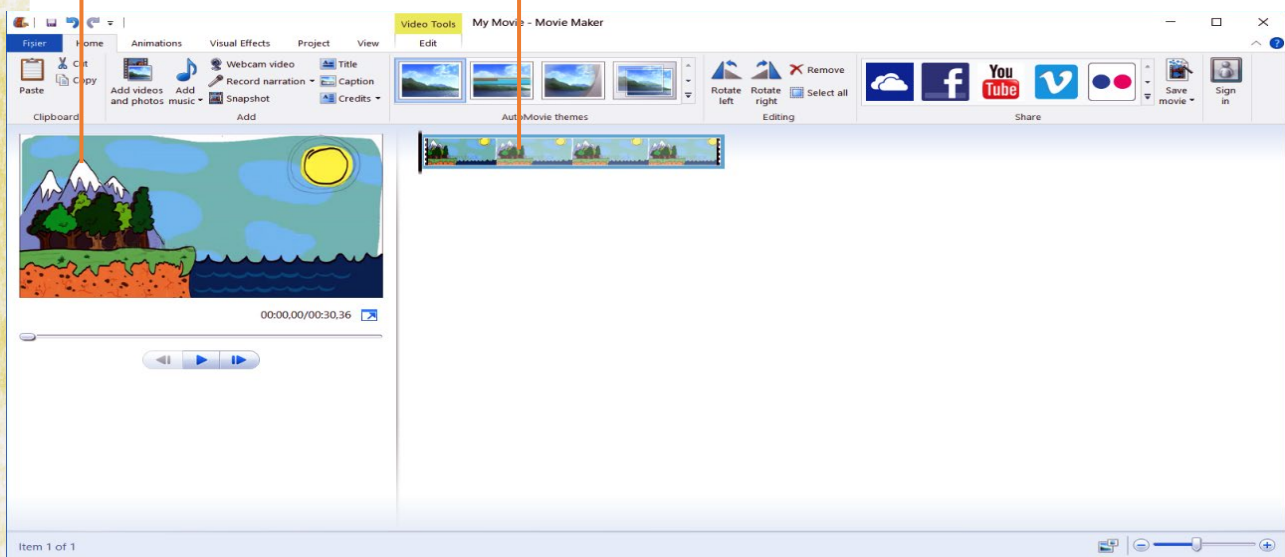


⊗ az első film a történelemben 1888. október 14-én jelent meg és 2 másodperces volt a hossza?

### 2.2.1. Audio-video állományokat feldolgozó alkalmazás kezelési felülete


A Windows Movie Maker alkalmazás fő feldolgozási részei az alábbiakban vannak leírva:

1. A kollázs megjelenítése      2. Képek és videofájlok





A „**Képek és videók**” részben kell hozzáadni a kollázshoz az összes kép vagy videó típusú fájlokat, amelyeket használni akarsz. Két lehetőség van ezen állományok hozzáadásához.

- Klikk a  gombra, aztán keresd meg, hova van mentve, és kattints az „Open”-re.
- A „drag and drop” módszert használva (mozgasd a fájlt az őt tartalmazó mappából az alkalmazás ablakának megfelelő részére).

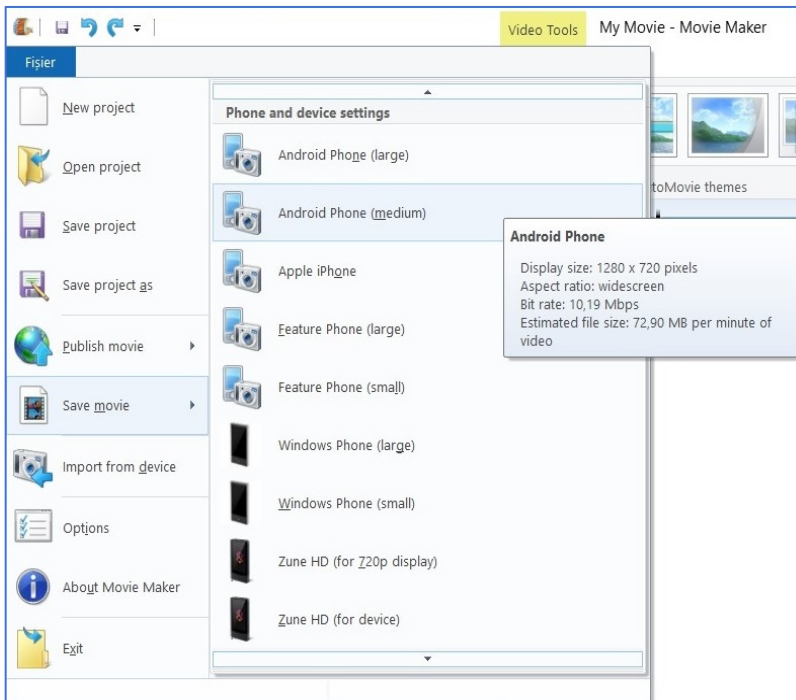
„**A kollázs megjelenítése**” lehetővé teszi, hogy lásd a videódat a létrehozás ideje alatt.

**Tudtátok, hogy...?**

- ⊗ egy videó típusú állomány *frame*-eknek nevezett képek sokaságából tevődik össze?
- ⊗ ez az állomány megengedi C++, HTML. stb. programozói nyelvekben használatos kódok szerkesztését?

## 2.2.2. Audio-video állományt feldolgozó alkalmazás műveletei

A kollázs mentése videofájlként a fő menüből történik kiválasztva a „File” → „Save movie” parancsot. A megjelenő párbeszéd ablak segít a videoállomány típusának beállításában.



A lejátszó eszköz függvényében különböző formátumok léteznek.

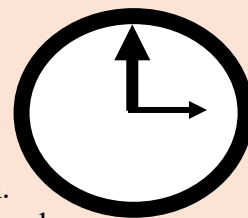
Amint a mellékelt ábrán is látható, ha hosszan rajtatartjuk a kurzort egy opción, akkor megjeleníthetők a tulajdonságok is.

**JEGYEZZÜK MEG!**

Ha később akarod folytatni a munkát, akkor menthatsz a „Save project” opcióval.

## Gyakorlatok

1. Nyitsd meg az audio-video fájl szerkesztésére alkalmas alkalmazást és adj hozzá a „Képek és videók”-hoz két a „Paint” alkalmazásban általad készített képet.
2. Exportáld az *enfilmem.mp4* néven általad szerkesztett videokollázst 1280\*720 felbontásban.
3. Egy videorögzítő készülék (pl. mobiltelefon) segítségével, készíts három, külön-külön maximum 30 másodperces interjúút osztálytársaidal. Az interjú témája „*Informatika és IKT óra*”. Mentsd el ezt a kollázst *sajattemam.mp4* névvel 2048\*1080 felbontásban.
4. A Paint alkalmazás segítségével készíts több órát ábrázoló rajzot, amelyek sorra 3:00, 3:10, 3:20, 3:30, 3:40, 3:50, 4:00 órát mutat. A Movie Maker alkalmazás segítségével készíts egy videoklipet ezeknek a képeknek felhasználásával a létrehozás sorrendjében. Mindegyik kép 1 másodpercig tartson a videoklipen belül. Mít veszel észre?



\*

Az osztálytársaidal és tanáraidal készített videók oktatási projektben való felhasználásához kérd az érintettek beleegyezését. Erre a személyi adatok védelmére vonatkozó törvények miatt van szükség.

## Egyéni feladat

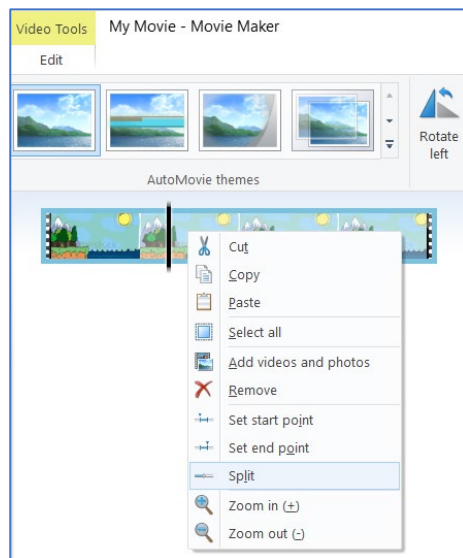
Alakíts ki egy 3-4 tagból álló csapatot és készítsetek egy tervezetet, amelyet majd bemutattok az osztály előtt, a következő témák közül:

- ⊗ A filmező kamera története;
- ⊗ A fényképezőgép története;
- ⊗ Egy filmtekercs előhívásának folyamata.

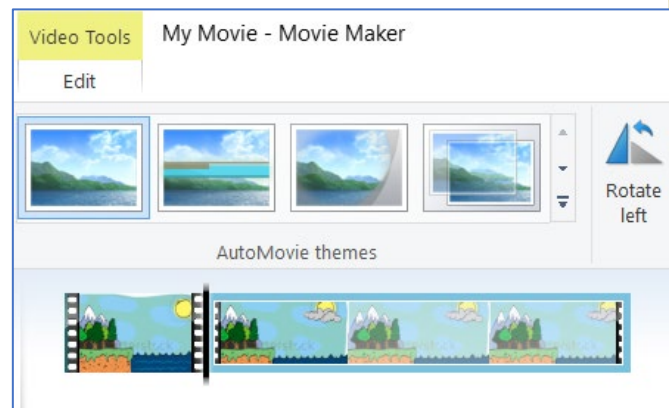
## VIDEOÁLLOMÁNYOK SZERKESZTÉSE

Megtörténik néha, hogy ki akarunk törölni néhány filmrészletet egy videó állományból. Ennek az eléréséhez, első lépésben el kell különítenünk ezt a videórészt. A „Split” gomb kettéoszt egy videó fájlt. Először elhelyezzük a fekete kurzort arra a helyre, ahonnan vágni akarunk, majd jobb klikk a változtatni kívánt videoklipen, a szövegre vonatkozó menüből pedig kiválasztjuk a „Split” opciót.

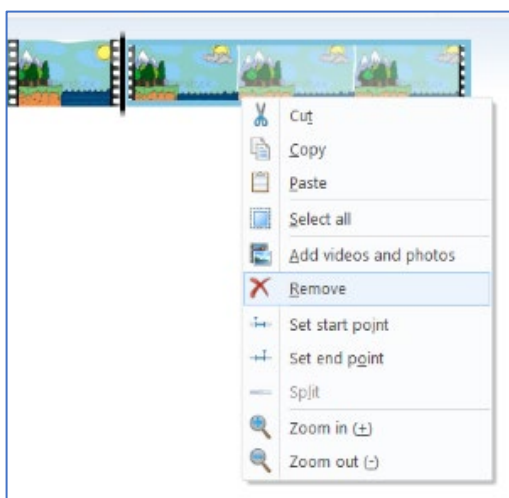
Előtte



Utána



Most már van két videórészletünk. Az első részlet az, amelyet szeretnénk megtartani a kollázsunkban, a második pedig amelyet szeretnénk eltávolítani.

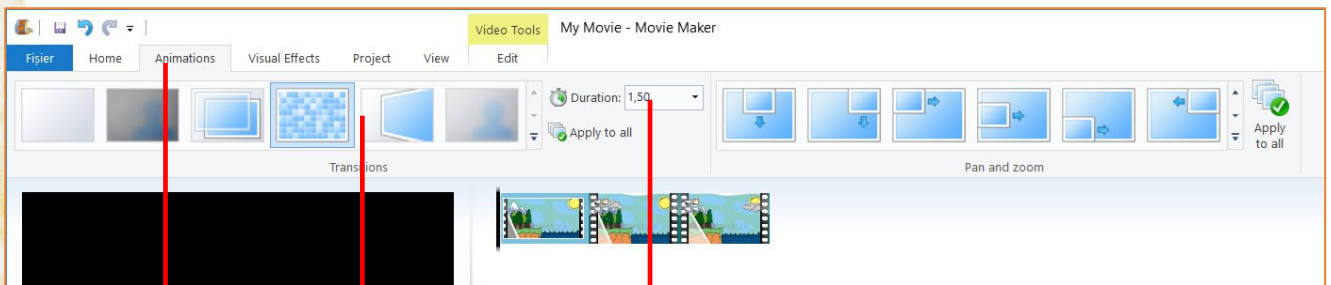


Egy videó kollázs valamely részének törlése a „Remove” gombbal történik. Ennek az opciónak az eléréséhez ugyanazokat a lépéseket kell megtennünk, mint a videoklip szétvágásánál. Jobb klikk a megfelelő videoklipen, aztán a helyi menüben megtalálhatjuk ezt az opciót.

Két videó fájl közötti átmenetet tulajdonképpen az egyik videó részletből a másikba való átváltás jelenti. Kiválaszthatjuk az átmenet módját a főmenü „ANIMATIONS” opciójának elérésével.

A kívánt átmenet módjának alkalmazásához csak ki kell választanod azt a videoklipet, amelyhez hozzá akarsz adni egy átmeneti effektust, aztán kiválasztod a kívánt effektust.





Animations  
opció

Átmeneti  
típusok

Az átmeneti időtartam

## Gyakorlatok

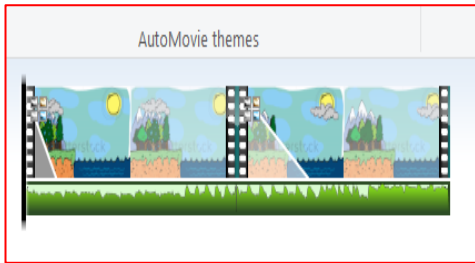
1. Nyiss meg szerkesztésre egy videó állományt, amelyet három egyenlő részre darabolsz. Miután eltávolítod a középső részt, adj hozzá egy átmeneti effektust a két megmaradó klip közé. Mentsd el a kapott állományt *feladat.mp4* néven.
2. Készíts egy videó kollázst az előző leckében elkészített három interjú alapján. Mielőtt exportárod *mp4* formátumba, vedd figyelembe a következőket:
  - távolítsd el az interjúkból a holtidőket úgy, hogy folyamatos beszéd legyen;
  - illessz be átmeneti effektusokat a három video közé.

\*

Az osztálytársaidal készített videókhoz kérünk, hogy kérd a tanárod és kollégáid beleegyezését egy tanítási projektben való felhasználáshoz. Ezekre a beleegyezésekre az érvényben levő személyi adatok védelme miatt van szükség.

## HANGSÁV ÉS SZÖVEGELEMEK BEILLESZTÉSE

Ahhoz, hogy hozzáadj egy hangszávet a kollázshoz, ajánlott, hogy legyen előkészítve az audiófájl. Ahhoz, hogy hozzáadd a projektedhez a kívánt hangállományt, az „Add music” gombot kell aktiválnod, amely megnyit egy kis ablakot és az egyik opció épp az „add music”. A következő lépés az, hogy keresd meg a számítógépen a kívánt állományt.



A kiválasztott állományt az alkalmazás alsó részéhez adjuk hozzá, a videó részletek alá (a mellékelt képen levő zöld szegély). Ha ki akarjuk vágni ezt a hangállományt, ugyanúgy járhatunk el, mint egy videó részlet kivágásakor. Elhelyezed a kurzort (fekete vonal) a kivágási részbe és aktiválsz a „split” gombot a jobb klikknek megfelelő menüből. Vigyázz, hogy a kivágási művelet előtt legyen kiválasztva az hangállomány. Megfigyelheted a lépéseket az alábbi képeken:

**1.**

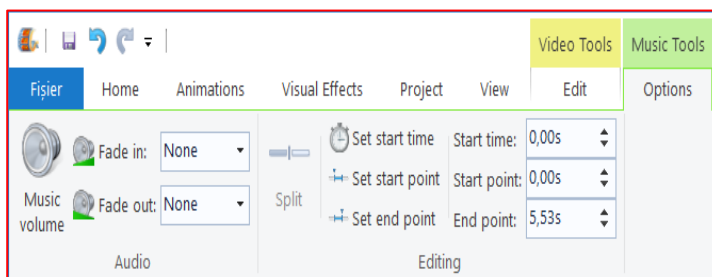
**2.**

**3.**

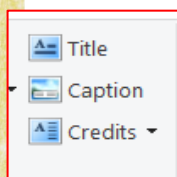
**4.**

Miután eldöntöttük, hogy a hangállomány mely részletét szeretnénk használni a videoklipünkben, áthelyezhetjük a „drag and drop” módszerrel a hangsávot arra a helyre, ahonnan szeretnénk, hogy kezdődjön.

A *Music Tools*-ban levő *Options* a *Music Tools*-ból nyitja meg a *Szalagot*, amely tartalmazza a hangerősség szabályozót, valamint a kezdő (Fade in) és befejező (Fade out) effektusokat. Ugyancsak innen tudjuk megadni a pontos indító, valamint befejező hangsáv koordinátákat.

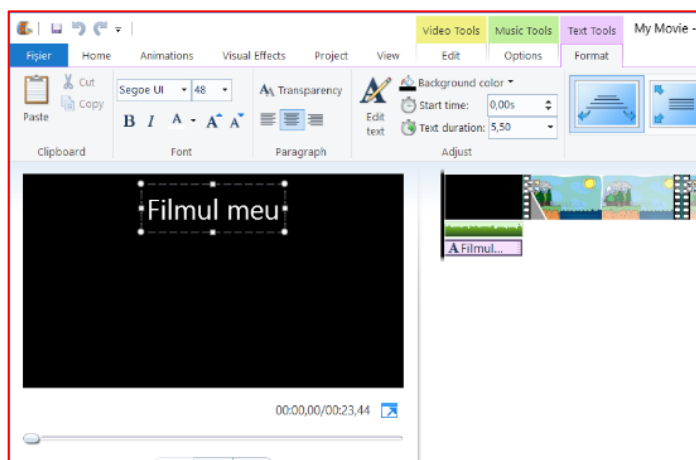


Lehetőségünk van a videó kollázshoz hozzáadni szöveget is. Ezt különböző formában tehetjük meg: bevezető szöveg, főcím vagy felirat. Mindezen opciók egyben megtalálhatók a főmenüben.

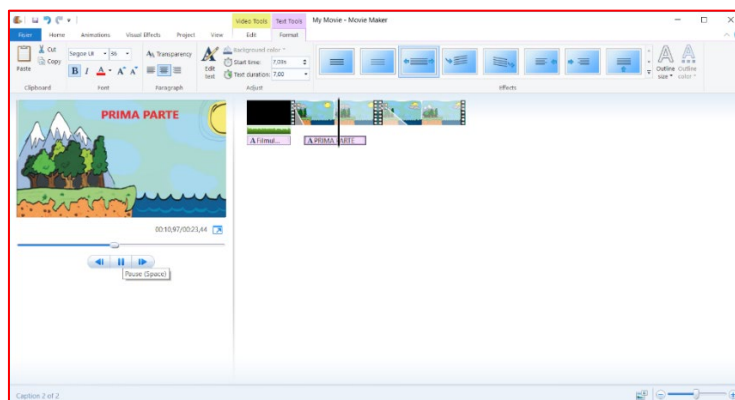


A „Title” abban segít, hogy hozzáadj egy címet a videoklip kezdetéhez:

a jobboldalon jelenik meg az a szövegdoboz, amelybe beírhatjuk azt, amit szeretnénk megjeleníteni a videoklip elején.

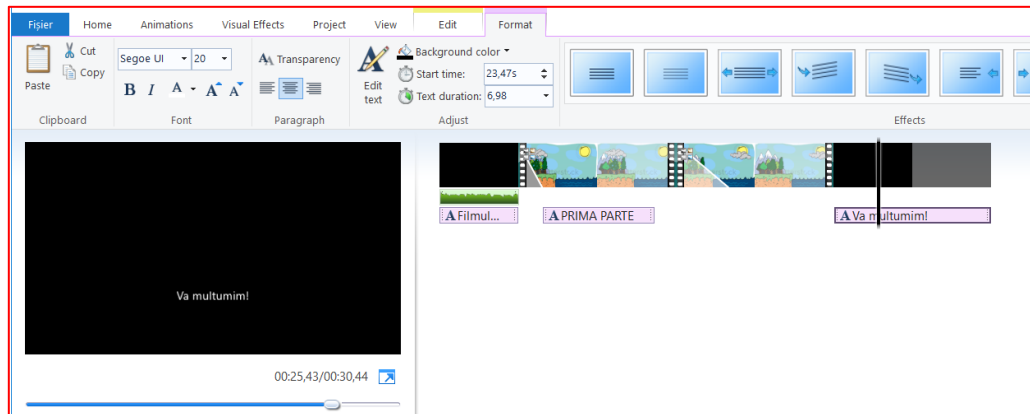


A „Caption” adja hozzá a kívánt szöveget a videoklip lejátszása alatt. Megjegyeznénk, hogy elég nehéz megvalósítani egy videoklip feliratát ezzel a módszerrel.

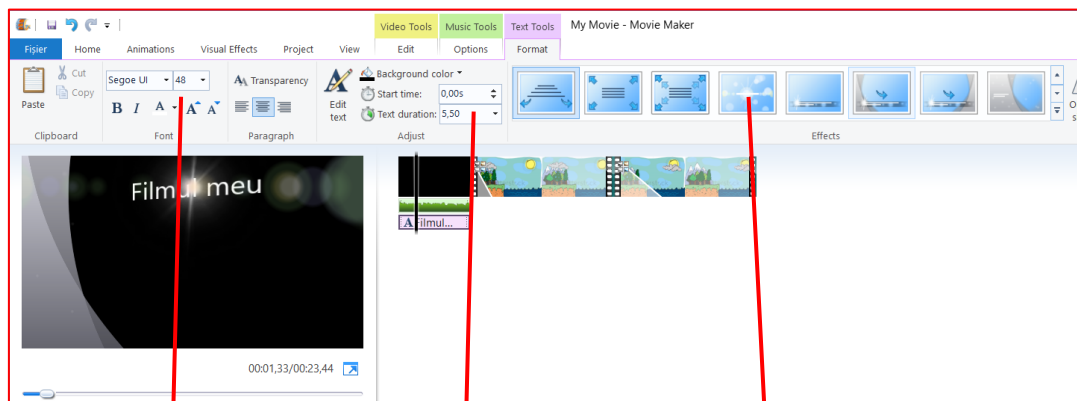




A „Credits” a „Title” szimmetrikusa. Általában a film elkészítésében résztvevők neveit tartalmazza. Az elkészített videokollázs végéhez kell hozzáadni.



A „Text Tools”-ban lévő „Format” tartalmazza az összes szükséges opciót a videó kollázs magas szintű elkészítéséhez:

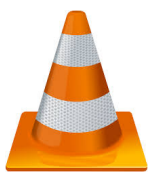


A betűk formázása.  
Pontosan úgy, mint  
Microsoft Word-ban.

A bevezető szöveg  
lejátszási ideje.

A megjelenítés módja. A képen  
megtalálható egy, már a  
bevezető szöveghez használt  
effektus is.

A videó állományok lejátszása különböző alkalmazások segítségével történhet meg, mint:



VLC Media Player



BS Player

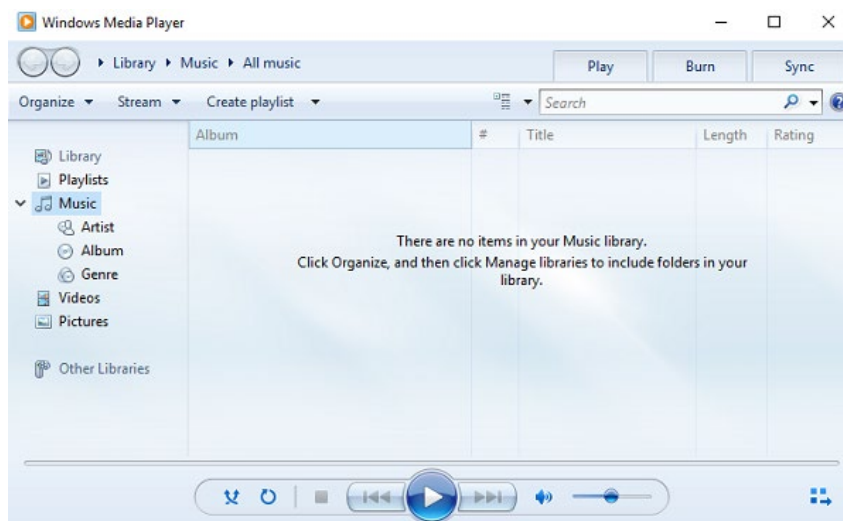


Windows Media Player

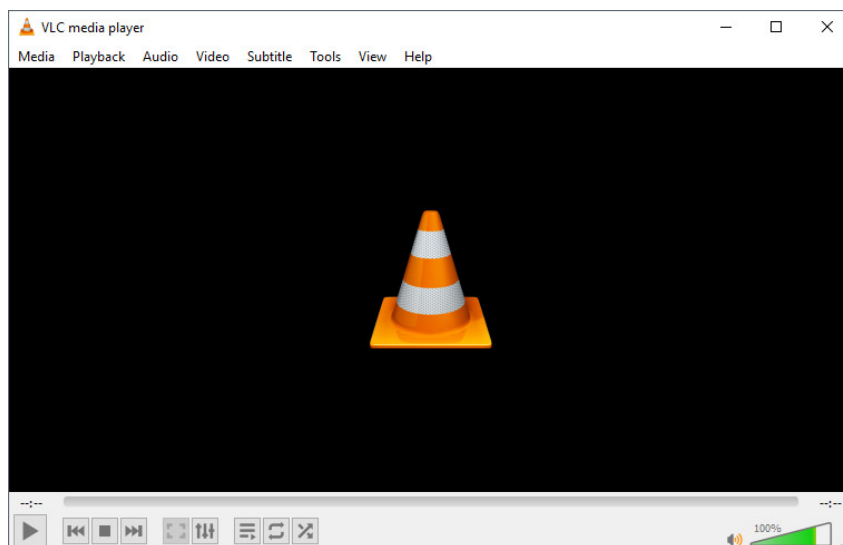
## A BS Player alkalmazás felhasználói felülete



## A Windows Media Player alkalmazás felhasználói felülete



## A VLC alkalmazás felhasználói felülete



## Gyakorlatok

1. Próbáld meg az előzőekben elkészített *feladat.mp4* videokollázshoz hozzáadni két hangsávot. Mindegyik hangállomány időhossza legyen a videofájl össziidejének a fele.
2. A *feladat.mp4* videoállomány kezdetéhez adj hozzá egy címet, amelynek tartalma az „Én kollázsom”, és a video lejátszásának ideje alatt a jobb felső sarokban jelenjen meg a neved.
3. Egy video rögzítő készülék segítségével (pl. mobiltelefon) készíts három, maximum 30 másodperces interjút néhány osztálytársaddal. Az interjú témája legyen „*Informatika és IKT óra*”. Mentsd el a kollázst *hazifeladatom.mp4* néven 2048\*1080 felbontásban.
4. Illeszd be a film végére az interjú alanyaira vonatkozó szereposztást.
5. A 3. pontban készített videóban írsd ki a résztvevő tanulók nevét és osztályát.

## Egyéni feladat

Alakítsatok ki egy 4-6 kollégából álló csapatot és készítsetek egy maximum 10 perces filmet „*Egy nap egy diák életében*” témával. Próbálgátok meg betartani a következő célkitűzéseket:

- ⊗ egy kezdő leírás, amelyben megjelenik a film címe;
- ⊗ egy végső leírás, amelyben a megjelenő személyek nevei jelennek meg;
- ⊗ minimum 4 videorészlet, amelyeket egy átviteli effektussal kell összekötni;
- ⊗ olyan hangsáv beillesztése, amely nem hallszik a párbeszéd alatt;
- ⊗ exportáljátok a kész projektet *.mp4* típusú formátumban;
- ⊗ tanári engedéllyel mutassátok be az elkészült filmet Informatika és IKT órán.

\*

Az osztálytársakkal készített videókhoz kérünk, hogy kérd a tanárod és kollégáid beleegyezését egy tanítási projektben való felhasználáshoz. Ezekre a beleegyezésekre az érvényben levő személyi adatok védelme miatt van szükség.





1. Kösd össze a baloldali oszlopban lévő elemeket a jobboldali megfelelőekkel: **1 p.**



Pause



Start recording



Stop

2. Egészítsd ki a következő mondatokat:

a. A videó kollázs mentését az alkalmazás **File** menüjében található .....  
opcióval tehetjük meg. **1 p.**

b. Az „**Add video and photos**” opció abban segít, hogy .....  
videoklipeket és képeket, amelyeket a kollázsban használsz. **1 p.**

3. Az alábbiak közül, melyik töröl egy videó részletet? **1 p.**

a. Split      b. Remove      c. Credits      d. Cut

4. Az alábbiak közül, melyik illeszt be egy kezdő feliratot? **1 p.**

a. Add music      b. Caption      c. Credits      d. Title

5. Válaszd ki a megfelelő választ:

A videofájlok lejátszására alkalmas applikáció: **1 p.**

a. Microsoft      b. Windows      c. Microsoft      d. Paint  
PowerPoint      Media      Word  
Player

6. Az alábbiak közül, melyikkel vághatunk ki (darabolhatunk fel) egy videorészletet? **1 p.**

a. Split      b. Remove      c. Cut      d. Title

7. Az alábbiak közül, melyikkel illeszthetünk be egy hangsávot? **1 p.**

a. Paste      b. Add music      c. Remove      d. Split

8. Kösd össze a baloldalon lévő elemeket a jobboldali megfelelőekkel: **1 p.**



BS Player



Windows Media Player



VLC Media Player

**Hivatalból jár:** **1 p.**

# **3. FEJEZET**

## **KOLLABORATÍV ALKALMAZÁSOK**

## A KOLLABORATÍV ALKALMAZÁSOK JELLEMZŐI. HOZZÁFÉRÉS/KAPCSOLÓDÁS A KOLLABORATÍV ALKALMAZÁSHOZ

Az eddigi Informatika és IKT tanórákon megtanultál prezentációkat készíteni PowerPoint-ban, szövegeket formázni Word-ben. Ezért a jövőben egyre több feladatot, csoportmunkát kaphatsz, amelyeket társaiddal közösen kell majd megoldj. Ezeket a feladatokat az Interneten keresztül is megoldhatjátok, anélkül, hogy szemtől- szembe lennétek egymással.



A kollaboratív alkalmazások segítenek, hogy csapatban dolgozhassatok az Internet segítségével. Tulajdonképpen a csapat információi online vannak tárolva és a csapat bármely tagjának van hozzáférése az állományokhoz, tudja bővíteni, tud törölni belőlük és módosíthatja tartalmukat. Az információkhoz való hozzáférés azonosító adatok segítségével történik.



Az egyik legelterjedtebb, állományok tárolását, szinkronizálását és szerkesztését lehetővé tevő szolgáltatás a Google Drive. A szolgáltatás online elérhető, a csatlakozás *felhasználónév* és *jelszó* segítségével történik. Ahhoz, hogy használni tudj, szükséged van egy Google *fiókra*.

Egy Google fiók létrehozása a <https://www.google.com/gmail/> webcím segítségével történik. Az oldalon a „Create an account” gombra kattintunk.

A képernyőn megnyíló ablakban, meg kell adni bizonyos adatokat, ahogy ezt a mellékelt kép is szemlélteti. Például:



- keresz- és családnevedet;
- egy felhasználó nevet (*username*) amelyhez, ha hozzáfűzöd a „*@gmail.com*” szöveget, máris létrejön az e-mail címed;
- a jelszót megerősítésként másodszor is be kell gépelni, azért, hogy meggyőződj, helyesen gépelted be első alkalommal;
- miután megbizonyosodtál, hogy a begépelte adataid helyesek, kattints a „*Next*” gombra.

**FIGYELEM!**

Egy biztonságos jelszó a következő követelményeknek kell hogy eleget tegyen:

- ✓ legkevesebb 8 karaktert kell tartalmazzon;
- ✓ tartalmaznia kell kis- és nagybetűt, számjegyet és speciális karaktert, mint: *! @ # \$ % ^ & \* ( ) \_ + = - ] [ { } | ' " ; : , . < > / ? - " ;*;
- ✓ mások számára ne legyen könnyen megjegyezhető;
- ✓ egyedi kell legyen – különböző fiókokhoz használj különböző jelszavakat;
- ✓ személyes adataiddal ne legyen kapcsolatban – ne tartalmazza a neved, születési dátumodat, vagy hasonló adatokat.

A képen látható ablak a következő adatokat kéri tőled:


- a telefonszámod; előtte ki kell válaszod az országot
- egy alternatív e-mail címet arra az esetre, ha elfelejtet a jelszavad;
- a születési dátumodat és a nemedet.


Ha többet szeretnél tudni a beírt adatok fontosságáról, kattints a „*Why we ask for this information*” szövegre.

Végül kattints a „*Next*” gombra.

Google

email, welcome to Google

 aplicatiicolaborative@gmail.com

 Phone number (optional)

We'll use your number for account security. It won't be visible to others.


Recovery email address (optional)

We'll use it to keep your account secure

Month  Day  Year

Your birthday

Gender



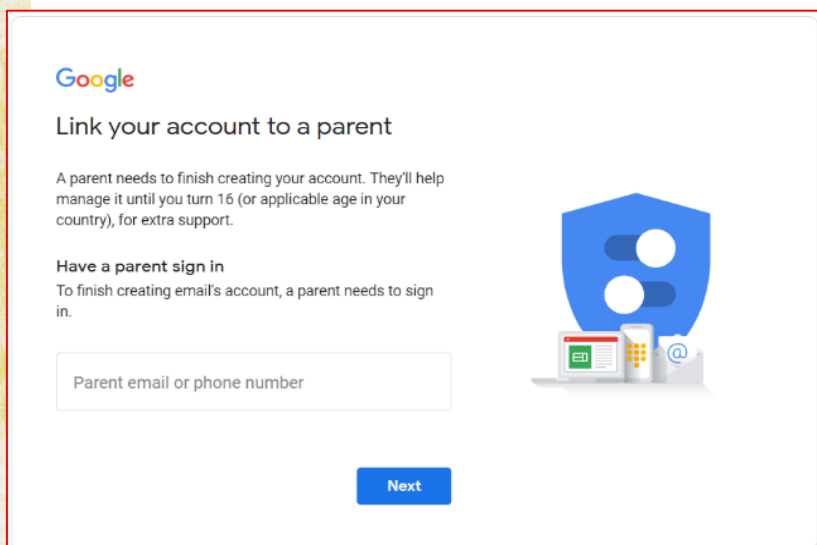
Your personal info is private & safe

[Why we ask for this information](#)

[Back](#) [Next](#)

## Tudtátok, hogy...?

- ⊗ van egy másik módszer is az elfelejtett Google-jelszó visszaszerzésére, mégpedig mobiltelefon segítségével?



Google

### Link your account to a parent

A parent needs to finish creating your account. They'll help manage it until you turn 16 (or applicable age in your country), for extra support.

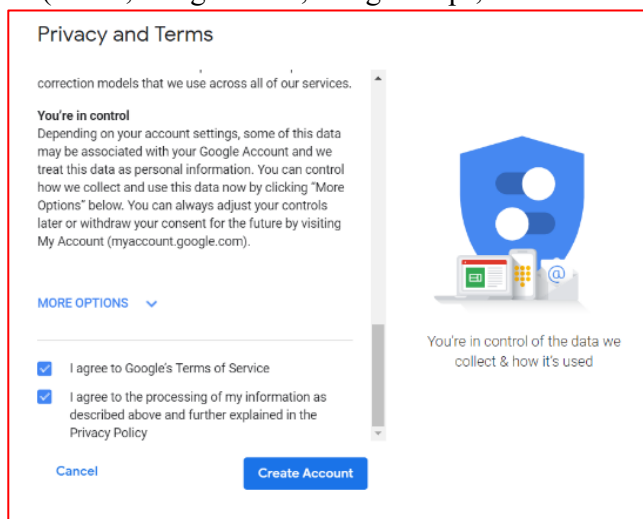
**Have a parent sign in**  
To finish creating email's account, a parent needs to sign in.

Next

Mivel még nem vagy 18 éves, ezért a fiók létrehozásához szükséges egy felnőtt engedélye. Ennek következtében a „**Parent e-mail or phone number**” mezőbe az egyik szülőd vagy nevelőd e-mail címe vagy telefonszáma kerül. Kérd meg a szüleid vagy nevelőd, hogy ellenőrizze postafiókját, ha az e-mail címet adta meg vagy olvassa el a kapott üzenetet, ha a telefonszám lehetőséget választottad.

A következő lépés, hogy elolvassd a titoktartási nyilatkozatot (Privacy and terms). Ha elolvastad figyelmesen és egyetértesz az olvasottakkal, válaszd ki a két lehetőséget a végén és nyomd meg a „Create Account” gombot.

Most már a végére értél a *gmail* fiók elkészítésének. A létrehozott fiók használható az összes Google szolgáltatás elérésére. (Gmail, Google Drive, Google Maps, YouTube etc.)



### Privacy and Terms

correction models that we use across all of our services.

**You're in control**  
Depending on your account settings, some of this data may be associated with your Google Account and we treat this data as personal information. You can control how we collect and use this data now by clicking "More Options" below. You can always adjust your controls later or withdraw your consent for the future by visiting My Account (myaccount.google.com).

**MORE OPTIONS** ▾

I agree to Google's Terms of Service

I agree to the processing of my information as described above and further explained in the Privacy Policy

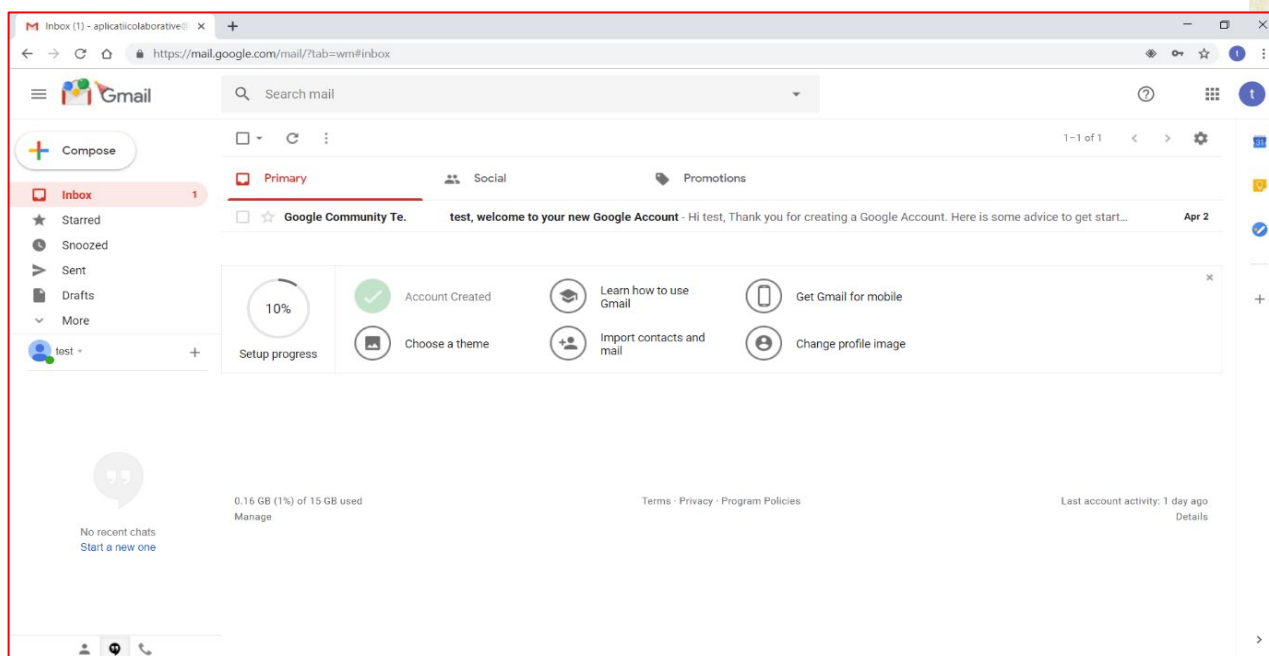
Cancel **Create Account**

You're in control of the data we collect & how it's used

## Tudtátok, hogy...?

- ⊗ egyes mobiltelefonok vagy táblagépek lehetővé teszik Google-fiók csatolását, ami folyamatosan elérhetővé teszi a dokumentumaidat, mivel online is mentve lesznek?

A Google fiók létrehozása után beléphetsz és megnézheted, hogy néz ki a gmail elektronikus postafiók.



Válassz megfelelő felhasználónevet az e-mail címhez, illetve a Google-fiókhoz. Gondolj arra, hogy aki kapcsolatba fog lépni veled a Google-on keresztül, tisztában kell legyen, hogy kivel kommunikál.

## Gyakorlatok

1. Hozz létre egy *gmail* címet, amely segítségével elektronikus üzeneteket tudsz küldeni az osztálytársaidnak.
2. Hozz létre egy Word dokumentumot, amelynek tartalma legalább 6 osztálytársad e-mail címe.
3. Küldj elektronikus üzenetet 3 osztálytársadnak.

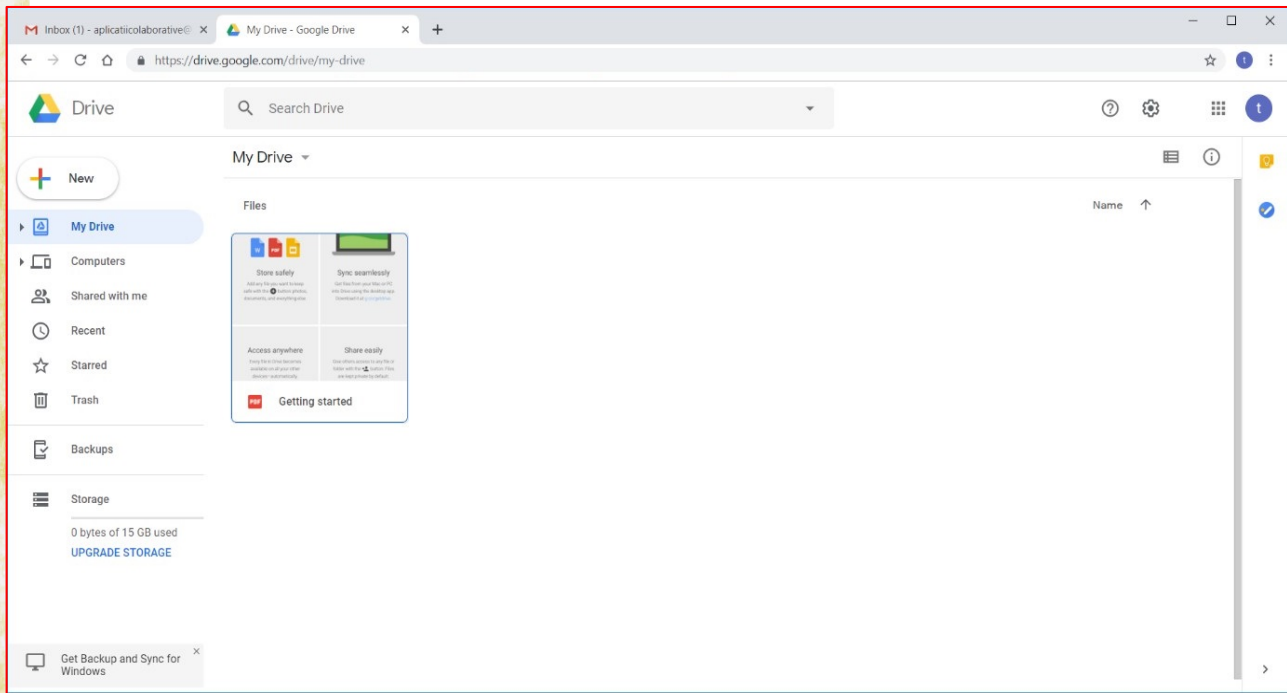
\*

A fenti követelmények megoldásához kérd a tanárod beleegyezését, valamint az osztálytársaidtól is kérj engedélyt az e-mail címük használatára. Ezekre a beleegyezésekre azért van szükség, hogy ne sértsd meg a személyes adatok védelmére vonatkozó jogszabályokat.

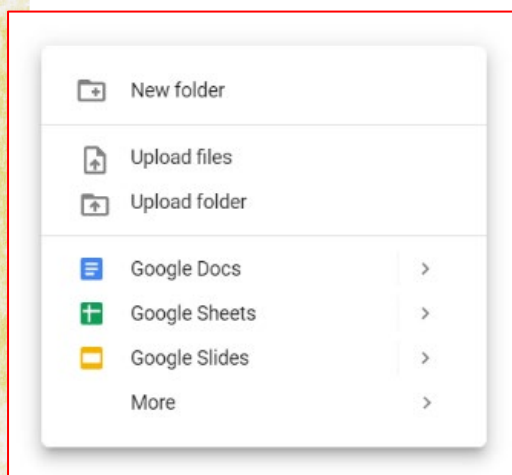


# KOLLABORATÍV ALKALMAZÁSOK LEHETŐSÉGEI ÉS FELHASZNÁLÓI FELÜLETE

A Google Drive alkalmazás használatához lépj a <https://www.google.com/drive/> oldalra.



A Google Drive segít az állományaid online tárolásában. Ezen az alkalmazáson keresztül más embereknek is engedélyezheted az állományaid megtekintését vagy szerkesztését.



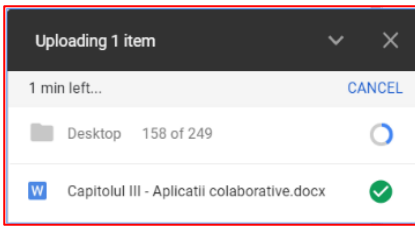
Az alkalmazás ablakában jobb klikk segítségével előjön a helyi menü, ahogy a mellékelt kép is mutatja.

Itt a következő lehetőségeid vannak:

**New folder** – új mappát hoz létre, az állományok és mappák észszerűbb rendezése és kezelése érdekében.

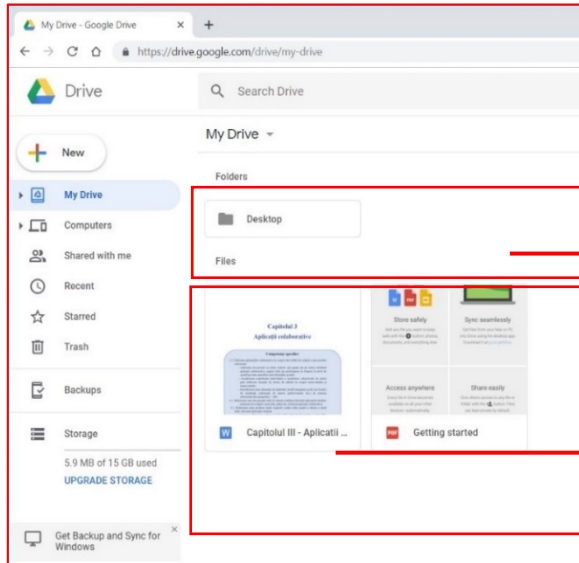
**Upload files** – egy párbeszédablakot nyit meg, ahol fel tudod tölteni a számítógépeden levő állományokat az online környezetbe.

**Upload folder** – egy párbeszédablakot nyit meg, ahol fel tudsz tölteni egy teljes mappát a számítógépedről az online környezetbe.



Miután kiválasztottad, hogy mit szeretnél feltölteni a Google Drive-ra, a jobb alsó sarokban megnyílik egy ablak, ahol láthatod, hogy mennyi idő alatt töltődnek fel az állományaid. Amikor az állomány teljesen feltöltődött, a zöld körben egy fehér pipa jelenik meg.

Az alábbi képen látható az állományok és mappák elrendezése:

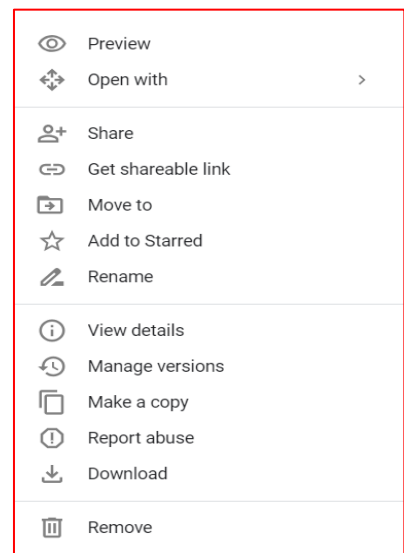


Itt található az összes általad létrehozott mappa. A mappák tartalmát dupla kattintással lehet megtekinteni.

Ezen a részen található az összes állományod.

Ha jobb egérgombbal kattintasz egy objektumra (állományra vagy mappára), előjön a helyi menü; amely a következő fő műveleteket tartalmazza:

- **Preview** – a kiválasztott állomány tartalmának megtekintése (csak állományokra érvényes);
- **Move to** – a kijelölt objektum áthelyezése máshova (csak a Google Drive-ban);
- **Rename** – a kiválasztott objektum átnevezése;
- **Make a copy** – a kijelölt állomány egy másolatának a létrehozása (csak állományokra érvényes);
- **Download** – egy objektum számítógépre való letöltése;
- **Remove** – a kijelölt objektum törlése.



## MUNKAESZKÖZÖK: DOKUMENTUMOK, BEMUTATÓK

Ebben a leckében megtanulod, hogy lehet a Word dokumentumokhoz hasonló állományokat (Google Docs) létrehozni, valamint a PowerPoint bemutatókhoz hasonló prezentációkat (Google Slides) készíteni.

VI. osztályban megtanultál bemutatókat készíteni PowerPoint-ban, az idén pedig szövegszerkesztést tanultál Word-ban. Ezért nem tárgyaljuk újra a menüpontok vagy gombok funkcionalitását.

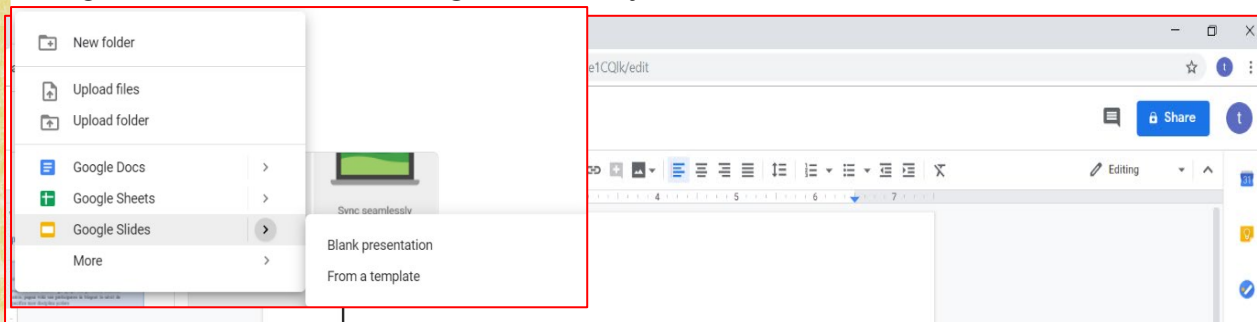
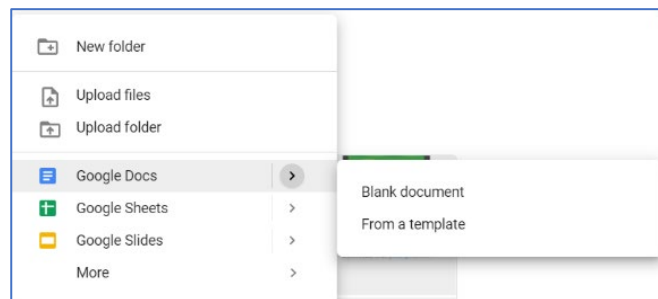
1. Hozz létre a Google Drive-on egy „*Informatika és IKT*” nevű mappát.
2. A létrehozott mappába töltsd fel a VII. osztályban informatika órán létrehozott összes állományodat.

### Gyakorlatok

#### Online Google Docs állomány létrehozása

Ugyanabból a menüből, ahol állományokat vagy mappákat tölthetsz fel, kiválaszthatod a Google Docs lehetőséget egy online állomány létrehozásához. Ahogy a mellékelt képen is látható, választhatsz egy üres dokumentum (Blank document) vagy egy előre definiált sablon (From a template) közül. Ha egy előre meghatározott sablont választasz, észreveheted, hogy a dokumentum formátumától függően különféle sablonok vannak (levél, önéletrajz stb.).

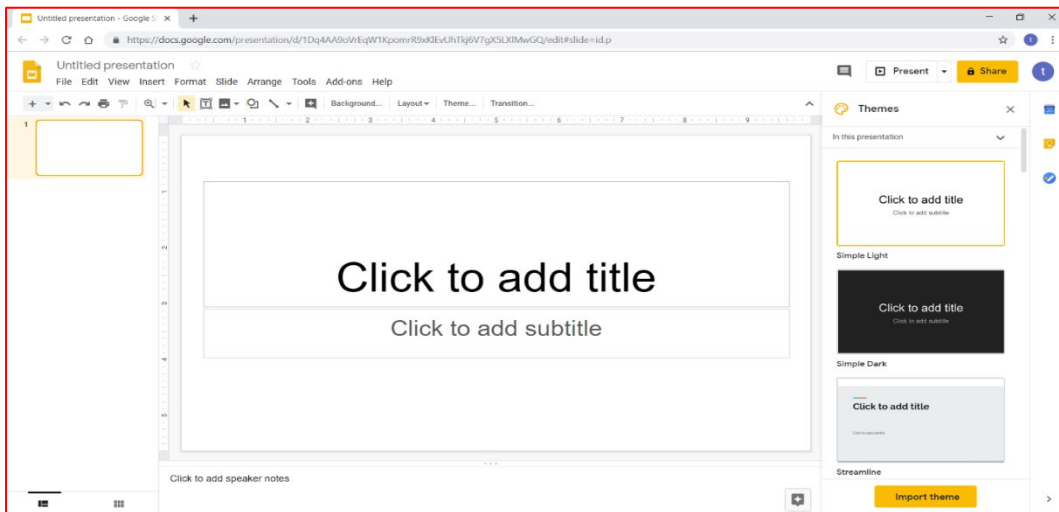
Az alábbi képen látható, hogy a Google Docs alkalmazás menüje nagyon hasonló a Microsoft Word alkalmazáséhoz. A gombok pontosan ugyanazzal a funkcióval rendelkeznek, mint amit már megtanultál és használtál a *Szövegszerkesztés* fejezetben.



#### Online Google Slides fájl létrehozása

Egy online bemutató létrehozható a Google Slides opcióval, a menüből. Ugyanazon opciók közül választhatsz, mint a Google Docs esetén. „Blank presentation” opciót választunk, ha egy üres bemutatót szeretnénk szerkeszteni, vagy a „From a template” lehetőséget, ha létező sablonnal szeretnénk dolgozni. A Google Docs menürendszere hasonlít a Microsoft Word szövegszerkesztő menürendszerére, míg a Google Slides a Microsoft PowerPoint applikációra hasonlít.

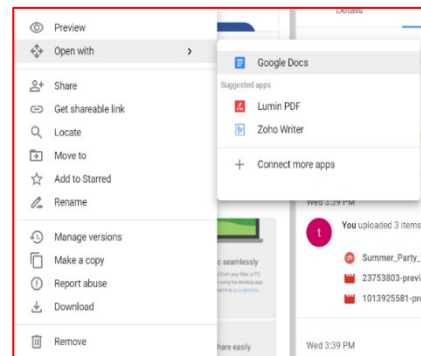




Egy másik lehetőség Google Docs dokumentum vagy Google Slides prezentáció létrehozására:

- létrehozod az állományt a számítógépeden;
- majd feltöltöd a Google Drive fiókodba;
- jobb klikk az állománynéven és kiválasztod az *Open with* lehetőséget, ahogy a mellékelt kép mutatja.

Fennáll a veszély, hogy a dokumentumban levő objektumok elrendezése megváltozik, különösen a képeké.



## Gyakorlatok

1. Készíts egy **Google Docs** állományt *osztálytársaim* néven. Az állomány az alábbi táblázatot tartalmazza:

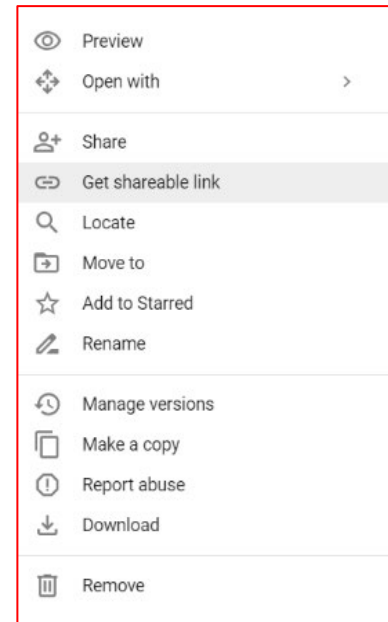
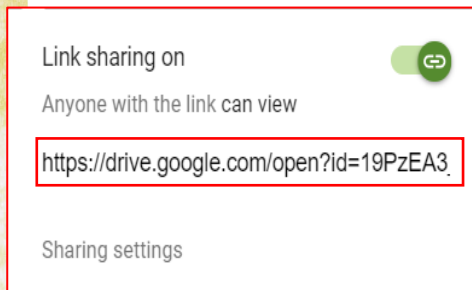
Vezetéknév	Keresztnév	Hobbik	Kedvenc tantárgyak

2. Készíts egy **Google Slides** állományt, amely segítségével bemutatsz az iskoládat (elhelyezkedés, hagyományok, eredmények, osztálytermek száma, tanárok stb.).
3. Hozz létre egy **Google Docs** állományt, amely tartalmazza az órarended.

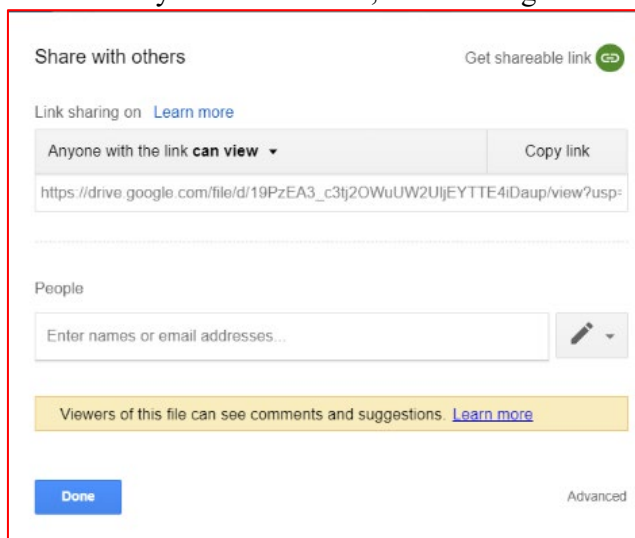
## MEGENGEDETT MŰVELETEK: KÖZZÉTÉTEL (PUBLIKÁLÁS), TARTALOM MÓDOSÍTÁSA

A kollaboratív alkalmazások érdekessége abban áll, hogy online dolgozhatsz a csapatod tagjaival anélkül, hogy feltétlenül szemtől-szemben ülnél velük. Ehhez a csapatodnak meg kell kapnia az állományod címét. Ez megtalálható, ha jobb egérgombbal kattintasz az objektumra, és a menüből kiválasztod a „*Get shareable link*” lehetőséget. A megjelenő ablakból lemásolhatod az állomány címét (az alábbi kép szövegdozozának a tartalma), és elküldheted a csapattársaidnak.

Abban a pillanatban, ahogy hozzáférnek az általad kapott címhez meg tudják tekinteni az állományodat anélkül, hogy tudnák módosítani (Anyone with the link can view).



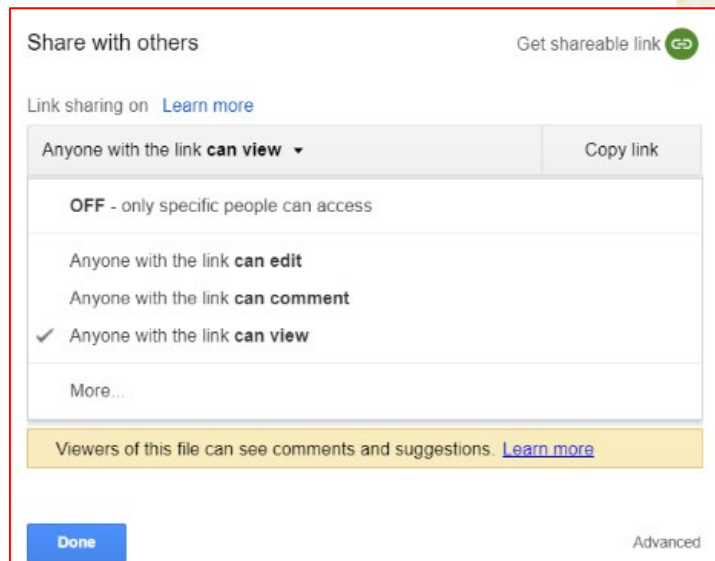
Annak érdekében, hogy módosításokat is tudjanak végezni az állományodon válaszd a „*Sharing settings*” opciót. Ennek az opciónak megfelelő párbeszédablak kétféle beállítást kínál fel. Az ablak felső részében be tudod állítani, hogy milyen jogokkal rendelkezzenek az adott címen levő állományhoz hozzáférő személyek. Az ablak alsó részében található szövegdozozban felsorolhatod, azon személyek e-mail címeit, akikkel megosztod az adott állományt, illetve a hozzáférés típusát.



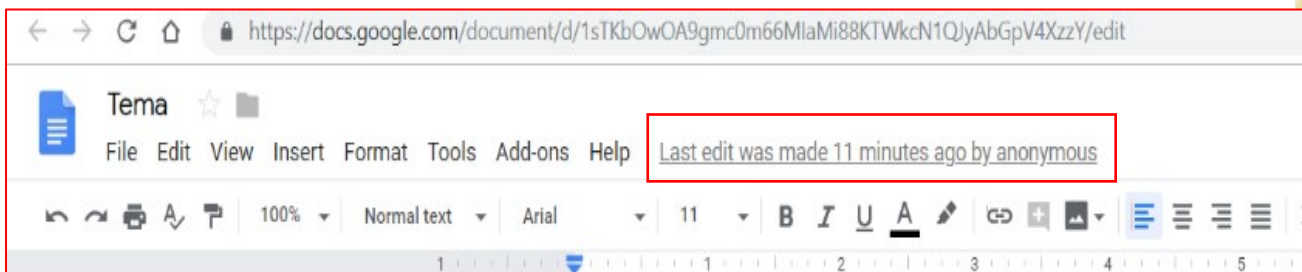
A személyeknek három különböző típusú jogot adhatunk:

- 1) **can edit** – tudja szerkeszteni a dokumentumot;
- 2) **can comment** – megjegyzéseket fűzhet a dokumentumhoz, anélkül, hogy joga lenne azt szerkeszteni;
- 3) **can view** – csak megtekinteni tudja a dokumentumot;

**OFF** – a hozzáférés nem megengedett.



A Google Drive menti a módosításokat, valamint az összes módosult állományváltozatot.

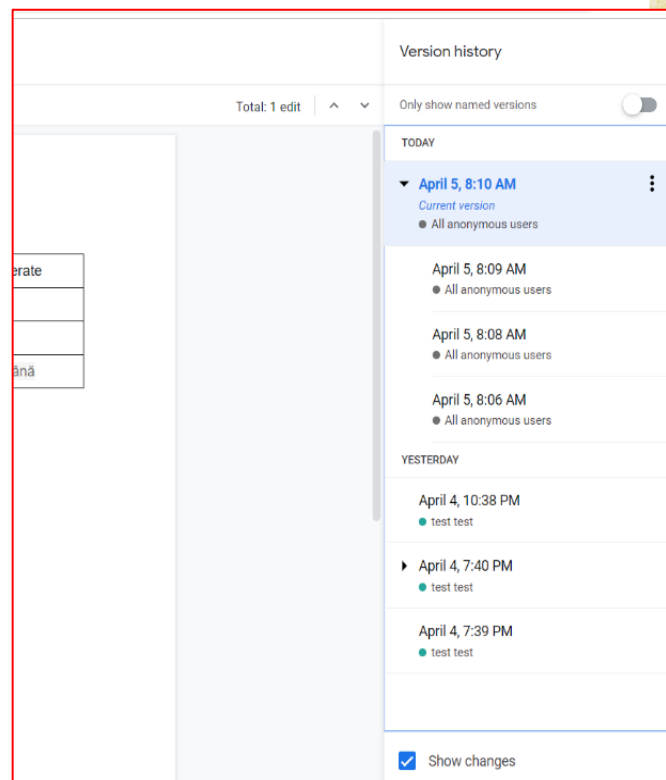
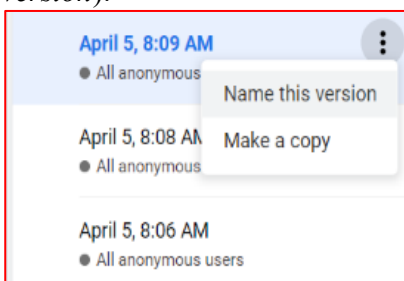


Ha rákattintasz az előző képen piros kerettel megjelölt részre megnézheted az állományon végrehajtott összes módosítást.

A képernyő jobb oldalán megnyílik egy ablak, amely tartalmazza minden változtatás dátumát és időpontját.

Az állomány bármelyik verzióját megtekintheted. Ha kiválasztod az egyik verzióját az állománynak, akkor megtekintheted ennek tartalmát és akár le is mentheted az addigi állapotában.

Nevet is adhatsz egy verzióknak (*Name this version*):





## FIGYELEM!

Amikor hálózaton keresztül dolgozol, védj személyes adataidat az illetéktelen hozzáféréstől. Ne tedd nyilvánossá a személyes adataidat, mint a címed, telefonszámod stb. Az illetéktelen személyek ellophatják a személyes adataidat és törvénytelen célokra használhatják.

Azokat a személyeket, akik a számítógépes hálózatokon keresztül olyan adatokat szeretnének elérni, amelyekhez nincs hozzáférésük hackereknek hívják.

A crackerek olyan személyek, akik egy alkalmazást vagy programot illegálisan használnak, anélkül, hogy megvásárolnák a szoftver licenzét. Ehhez módosítaniuk kell az alkalmazást úgy, hogy telepítéskor a biztonsági szint minimális legyen.

## Gyakorlatok

1. Adj szerkesztési jogot az *osztálytársaim* nevű állományra és küld el a webcímét az osztálytársaidnak, hogy egészítsék ki ők is a táblázatot.
2. Hozz létre egy online dokumentumot *mese* néven. Írj a dokumentumba egy 4-5 soros, általad kitalált meserészletet. Küld el a dokumentum webcímét az osztálytársaidnak, hogy ők is kiegészíthessék a történetet 4-5 sorral. Az így született mesét olvassátok fel az osztályban.

## PROJEKT

Alkoss csapatot 5-6 osztálytársaddal. Hozzatok létre egy Google Slides állományt, amely tartalmazza a következő diakockákat:

- ⊗ első dia – a csapat neve, osztály és iskola megnevezése;
- ⊗ a következő diákon - egy-egy csapattag bemutatása: egy profilkép minden csapattagról, a nevük és egy rövid bemutatása minden tagnak;
- ⊗ a többi dia – tartalmazza egy lecke bemutatását egy általatok választott tantárgyból.

# **4. FEJEZET**

## **PROGRAMOZÁSI NYELV**

## ISMÉTLÉS: GRAFIKAI BLOKKOKKAL ÁBRÁZOLT ALGORITMUSOK

A fejezet első leckéjében átismételjük az algoritmusok grafikus ábrázolásmódjáról tanultakat. Ez segíteni fog abban, hogy jobban megértsd és alkalmazd a C++ programozási nyelvhez kapcsolódó új ismereteket.

### A szereplő mozgatása egy útvonal mentén

Az algoritmus könnyebb alkalmazása céljából indokolt olyan szereplőt választani, amelynek a bal és a jobb oldala más-más színű. Ez segít abban, hogy megfigyeljük azt a részt, ahol megdől az útvonal.




A szereplő koordinátáinak kezdeti értékeket adunk, így tudni fogjuk, hogy minden esetben honnan indul az elmozdulás. Ehhez a mellékelt utasítástömböket fogjuk használni: **ugorj ide** és **nézz ... fokos irányba**.

Mivel nem ismerjük az útvonal hosszát, evidens, hogy ismétlődő utasításra lesz szükségünk.


Ennek az utasításnak a belsejében meg kell mondjuk, hogy lépjen előre, és ha elveszti az útvonalat, keresse meg.

**FIGYELEM!** Az elmozdulást ne végezzük túl gyorsan, mert a szereplő nagyon eltávolodhat az útvonaltól!

A szereplőnk három állapotban lehet:

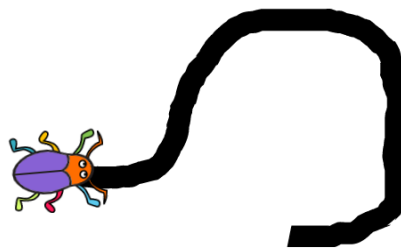
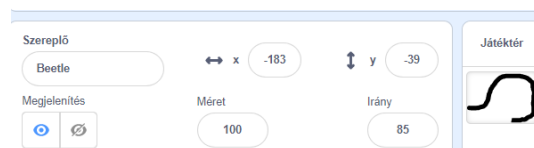
		
Az út egyenesen előre van. Ebben az esetben mehet előre.	Az út balra tér. Arra kell utasítani, hogy egy kicsit forduljon balra.	Az út jobbra tér. Arra kell utasítani, hogy egy kicsit forduljon jobbra.


**Megjegyzés!** A szereplőnk első lábainak színét felhasználva tudjuk megfigyelni, hogy az út valamilyen irányba kanyarodik.



```

- ra kattintáskor
ugorj ide: x: -183 y: -39
nézz 85 fokos irányba
mindig
  menj 2 lépést
  ha szín érinti fekete színt? akkor
    fordulj 5 fokot
  ha szín érinti fekete színt? akkor
    fordulj 5 fokot
  
```

Szereplő: Beetle  
 Megjelenítés:    
 Méret: 100  
 Irány: 85  
 Játéktér: 



Emlékezzünk vissza!

Egy **algoritmus** véges számú, rendezett lépések (utasítások, parancsok) sorozatát jelenti, amely ismert *bemeneti adatokból* megkapja a *kimeneti adatokat* (eredményt) *közbeeső adatokat* felhasználva, ha szükséges.

Az algoritmusok néhány tulajdonsága: véges, világos, általános.

Az algoritmusok adatokkal dolgoznak. Ezeknek van neve, típusa és értéke.

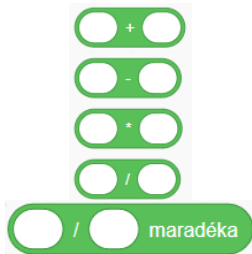
A kifejezés operátorokból és operandusokból áll.

Az operandusok lehetnek:

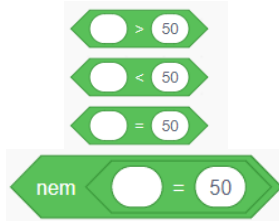
- konstansok;
- változók.

Az operátorok lehetnek:

aritmetikai



relációs



logikai



Példák kifejezésekre

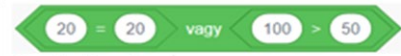
A kifejezés eredménye lehet:

- logikai (igaz/hamis);
- aritmetikai (numerikus).



Eredmény: hamis  
Megjegyzés: hamis és...

→hamis; nincs szükség kiértékelni az és utáni kifejezést, ha az első kifejezés hamis. Miért?



Eredmény: igaz

Megjegyzés: igaz vagy...  
→Igaz; nincs szükség kiértékelni a vagy utáni kifejezést, ha az első kifejezés igaz. Miért?

Emlékezzünk vissza!

Az algoritmusok leírásának egyik egyszerű és érthető ábrázolásmódja a **grafikus blokkokkal** való ábrázolás. Ezek a következők:

### a) szekvenciális (lineáris) struktúrák

<i>olvasás</i> – átvesz egy értéket és a változónak adja.		Beolvassa az <b>a</b> változó értékét
<i>írás</i> – kiírja egy változó tartalmát, vagy egy üzenetet		Kiírja az <b>a</b> változó tartalmát
		Kiírja a <b>helyes</b> (szöveges) üzenetet
<i>értékdás</i> – egy változó értéket kap; az		Az <b>x</b> változó értéke <b>3</b> lesz

érték egy kifejezés eredménye is lehet		Az <b>y</b> változó értéke az <b>x+2</b> kifejezés eredménye lesz
		Az <b>n</b> értéke csökken 1-el

### b) elágazási (döntéshozatal) struktúrák

	Meghatározza a <i>logikai_érték</i> igazságértékét. Ha ez <i>igaz</i> , akkor végrehajtja az <b>utasítás 1</b> , műveletsort, ha <i>hamis</i> akkor az <b>utasítás 2</b> műveletsort.
	Meghatározza a <i>logikai_érték</i> igazságértékét. Ha ez <i>igaz</i> , akkor végrehajtja az <b>utasítások</b> , műveletsort, ha <i>hamis</i> akkor kilép a struktúrából.



#### 1. Tekintsük a következő algoritmust:

```

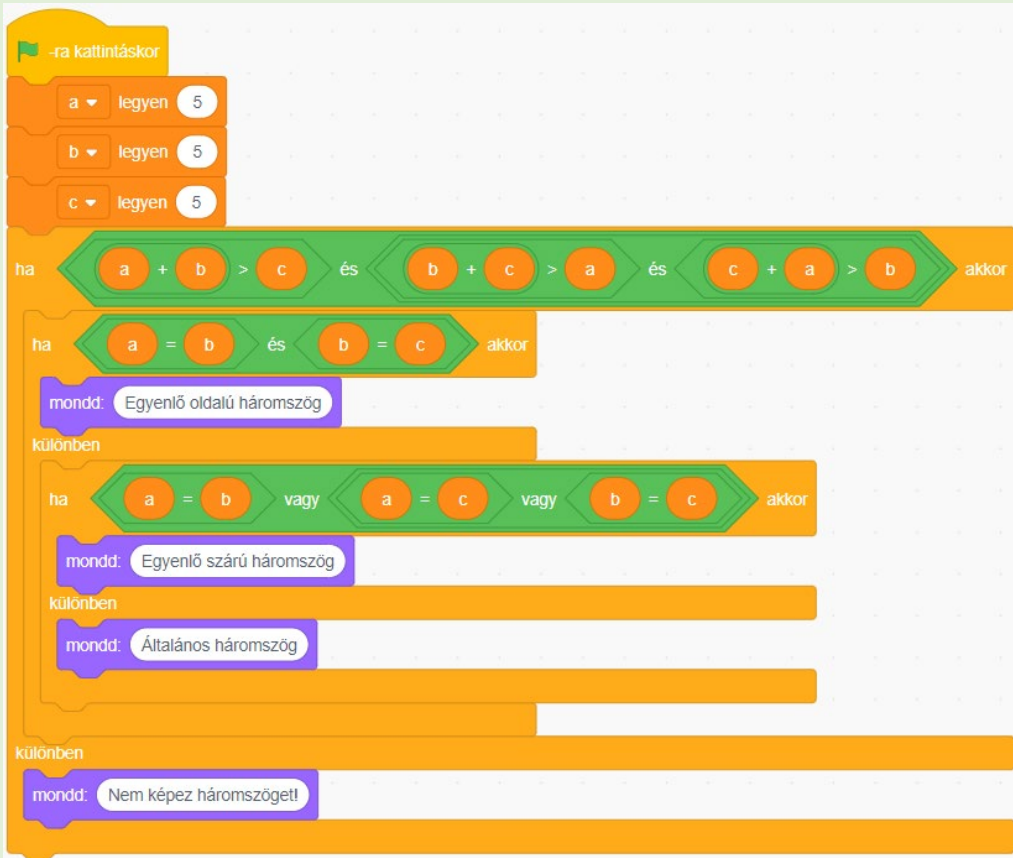
-ra kattintáskor
  a legyen 4
  b legyen 2
  c legyen 8
  ha (a + b > c és b + c > a és c + a > b) akkor
    mondd: háromszög
  különben
    mondd: nem háromszög
  
```

Kövessd a végrehajtás lépéseit, ha

- í.  $a=5, b=7, c=10$
- íí.  $a=4, b=2, c=8$

í.  $5 + 7 > 10$  és  $5 + 10 > 7$  és  $7 + 10 > 5$ ?  
Igen → A képernyőn megjelenik a **háromszög** szöveg.  
íí.  $4 + 2 > 8$  és...? Nem  
Mivel az első logikai kifejezés *hamis*, a logikai operátor pedig az **és**, nem szükséges a többi kifejezés kiértékelése.  
→ A képernyőn megjelenik a **nem háromszög** szöveg.  
Adj meg olyan **a**, **b** és **c** értékeket, amelyre a **háromszög** szöveget írja ki.

2. Adott a következő algoritmus:



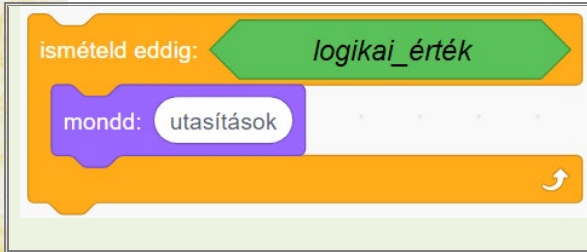
Kövessd a végrehajtás lépéseit, ha

- í.  $a=5, b=5, c=5$
- íí.  $a=4, b=4, c=6$
- ííí.  $a=4, b=3, c=6$

í.  $5 + 5 > 5$  és  $5 + 5 > 5$  és  $5 + 5 > 5$ ?  
Igen  
Kiértékeli  $5 = 5$  és  $5 = 5$ ?  
Igen → A képernyőn megjelenik az **Egyenlő oldalú háromszög** szöveg.  
íí.  $4 + 4 > 6$  és  $4 + 6 > 4$  és  $4 + 6 > 4$ ?  
Igen  
Kiértékeli  $4 = 4$  vagy ...? Igen  
Az első kifejezés logikai értéke *igaz* és a logikai operátor **vagy**, nem szükséges a többi kifejezés kiértékelése.  
→ A képernyőn megjelenik az **Egyenlő szárú háromszög** szöveg.  
ííí. Próbáld meg! Mi jelenik meg a képernyőn?



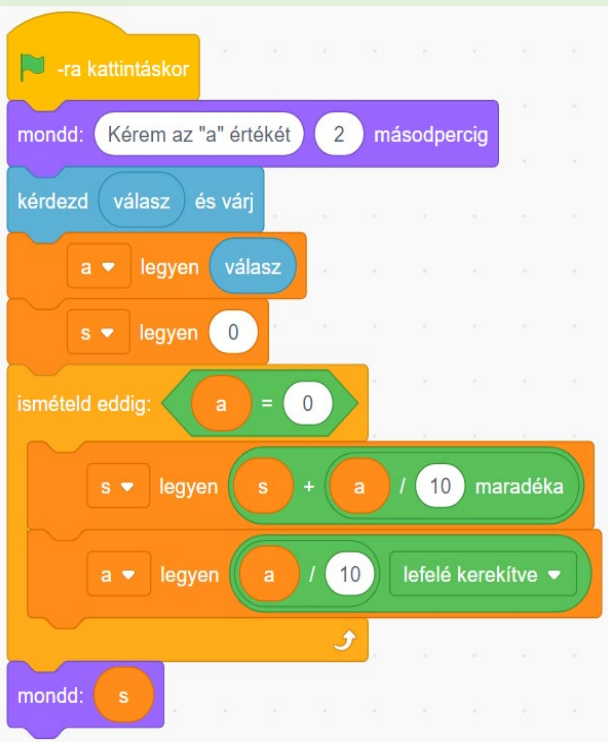
### c) ismétlő struktúrák



1. Meghatározza a *logikai\_érték* igazságértékét.
2. Ha *igaz*, akkor végrehajtja az **utasítások** blokkot, ha pedig *hamis*, kilép a struktúrából.
3. Ismétli az 1-es lépéstől.



3. Legyen a következő algoritmus:  
**a/b maradéka** az **a** természetes számnak a **b** nem nulla természetes számmal való osztási maradékát jelenti.



Fogalmazd meg az algoritmushoz tartozó feladatot!

Kövessd lépésről – lépésre a végrehajtását, ha **a=415**.

lépés	$a \neq 0?$	változók		képernyő
		$a$	$s$	
		415	0	
1	<i>igen</i>	41	5	
2	<i>igen</i>	4	5	
3	<i>igen</i>	0	10	
4	<i>nem</i>			10

4. Legyen a következő algoritmus:

**a/b maradéka** az **a** természetes számnak a **b** nem nulla természetes számmal való osztási maradékát jelenti.

Fogalmazd meg az algoritmushoz tartozó feladatot!

Kövess lépésről – lépésre a végrehajtását, ha **a=4712**

lépés	$a \neq 0?$	$a/10$ maradék >mszj	változók		képernyő
			a	mszj	
			4712	0	
1	igen	igen	471	2	
2	igen	nem	47		
3	igen	igen	4	7	
4	igen	nem	0		
5	nem				7

- íí. Módosítsd az algoritmust úgy, hogy kiírja:
- az **a** legkisebb számjegyét;
  - hányszor fordul elő egy **c** (beolvasott) számjegy az **a** számban;
  - a legnagyobb számjegyet, és hogy hányszor fordul elő.

5. Legyen a következő algoritmus:

**a/b maradéka** az **a** természetes számnak a **b** nem nulla természetes számmal való osztási maradékát jelenti.

Fogalmazd meg az algoritmushoz tartozó feladatot!

Kövess lépésről – lépésre a végrehajtását, ha **a=24, b=10**

lépés	$b \neq 0?$	változók			képernyő
		a	b	r	
		24	10		
1	igen	10	4	4	
2	igen	4	2	2	
3	igen	2	0	0	
4	nem				2

6. Legyen a következő algoritmus:  
 $a/b$  maradéka az  $a$  természetes számnak a  $b$  nem nulla természetes számmal való osztási maradékát jelenti.

```

- ra kattintáskor
mondd: Kérem az "a" értékét 2 másodpercig
kérdézd: válasz és várj
a legyen válasz
k legyen 0
ha a = 0 akkor
k legyen 1
ismételd eddig: a = 0
ha a / 10 maradéka = 0 akkor
k legyen k + 1
a legyen a / 10 lefelé kerekítve
mondd: k
  
```

Fogalmazd meg az algoritmushoz tartozó feladatot!

í. Kövesd lépésről – lépésre a végrehajtását, ha  $a=40100$

lépés	$a/10$ maradéka $= 0?$	változók			képernyő
		$k$	$a$	$a = 0?$	
		0	40100		
1	igen	1	4010	nem	
2	igen	2	401	nem	
3	nem		40	nem	
4	igen	3	4	nem	
5	nem		0	igen	3

íí. Módosítsd az algoritmust úgy, hogy kiírja:  
**a.** az  $a$  számjegyeinek számát;  
**b.** az  $a$  páros számjegyeinek számát;

7. Legyen a következő algoritmus:

```

- ra kattintáskor
a legyen 5
b legyen 10
s legyen 0
nr legyen 1
i legyen a
ismételd b = 1
s legyen s + nr * i
nr legyen nr * -1
i legyen i + 1
mondd: s
  
```

Fogalmazd meg az algoritmushoz tartozó feladatot!  
 Általánosíts!

í. Kövesd lépésről – lépésre a végrehajtását, ha  $a=5$  és  $b=10$

lépés	$i \leq b?$	változók					képernyő
		$a$	$b$	$s$	$nr$	$i$	
		5	10	0	1	5	
1	igen			5	-1	6	
2	igen			-1	1	7	
3	igen			6	-1	8	
4	igen			-2	1	9	
5	igen			7	-1	10	
6	igen			-3	1	11	
	nem						-3

íí. Tudsz írni egy olyan algoritmust, amely nem használ egyetlen ismétlődő struktúrát sem?



8. Legyen a következő algoritmus:  
**a/b maradéka** az **a** természetes számnak a **b** nem nulla természetes számmal való osztási maradékát jelenti.

```

-ra kattintáskor
a legyen 36
ismételd eddig: a / 2 maradéka > 0
a legyen a / 2 lefelé kerekítve
k legyen 0
i legyen 1
ismételd eddig: i > a / 2 lefelé kerekítve
ha a / i maradéka = 0 akkor
k legyen k + 1
i legyen i + 2
k legyen k + 1
mondj: k
  
```

Fogalmazd meg az algoritmushoz tartozó feladatot!

í. Kövesd lépésről – lépésre a végrehajtását, ha **a=36**.

Lépés	$i \leq a/2$ lefelé kerekítve?	$a/2$ maradéka = 0?	$a/i$ maradéka = 0?	változók			Képernyő
				a	k	i	
				36			
1		igen		18			
2		igen		9			
3		nem			0	1	
4	igen		igen		1	3	
5	igen		igen		2	5	
6	nem				3	3	

íi. Módosítsd az algoritmust úgy, hogy kiírja:

- a. az **a** páratlan osztóinak összegét;
- b. **a** osztóinak számát;
- c. azt, hogy az **a** prímszám vagy nem prímszám;
- d. az **a** páros osztóinak számát.

**Egyéni feladat**

1. Párosítsd a találó elemeket!

1.  $2 < 4 \cdot 5$
2. nem  $2 < 8$
3.  $7 / 3$  maradéka
4.  $2 + 3 \cdot 5$
5.  $123 / 10$  lefelé kerekítve

- a. 10
- b. hamis
- c. 1
- d. 12
- e. 17
- f. igaz

2. Egészítsd ki:

a. A  $2 * a + b / 5$  kifejezésben az operandusok: ....., és az operátorok: .....

b.  $2 < 3$  és nem  $4 = 3 * 6 / 2$  maradéka eredménye .....

c. nem  $7 < 11 / 2$  vagy  $5 = 3 + 2$  eredménye .....

3. Melyik kijelentés **igaz**, és melyik **hamis**?

a. Az állandók (konstansok) olyan adatok, amelyek leg több 3 alkalommal módosulnak az algoritmus végrehajtása során.

b.  $145 / 10$  maradéka eredménye 5.

c. Egy változóban nem lehet karakter típusú adatot tárolni.

4. Melyik kijelentés **igaz**, és melyik **hamis**?

a. Mindenik algoritmus tartalmaz egy lineáris struktúrát.

b. A  $m = n + 2$  utasításban az  $n$  változó értéke előre ismert kell legyen.

c. Az  $i$  változó kezdeti értékétől függetlenül a  $i = i + 2 - 1$  kifejezés eredményeként  $i$  csak páratlan lehet.

d. A `mondd: x` és a `mondd: x` utasítások között nincs különbség.

5. Egészítsd ki:

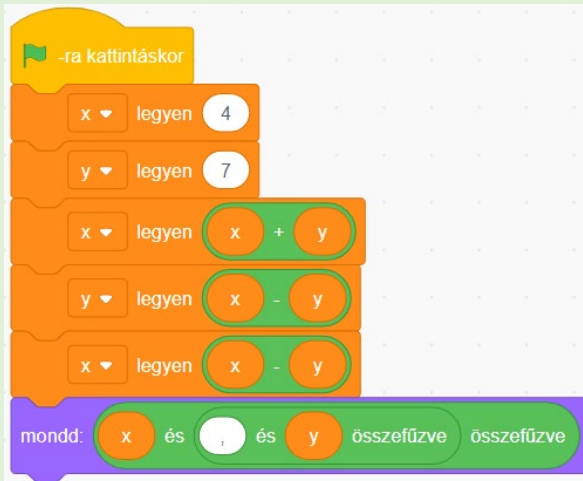
A `mondd: Az összeg: és s + 1 összefűzve` utasítás kiírja a képernyőre: .....

6. Kövesd végig a következő algoritmust lépésről - lépésre és határozd meg, mit fog kiírni, ha **a**-nak a beolvasott értéke 10, **b**-nek pedig 5.

```

- ra kattintáskor
kérdézd Kérem az "a" értékét és várj
a legyen válasz
kérdézd Kérem a "b" értékét és várj
b legyen válasz
a legyen a + 1
b legyen b - 2
mondd: a + b
    
```

7. Kövesd végig a következő algoritmust, és határozd meg, mit fog kiírni.

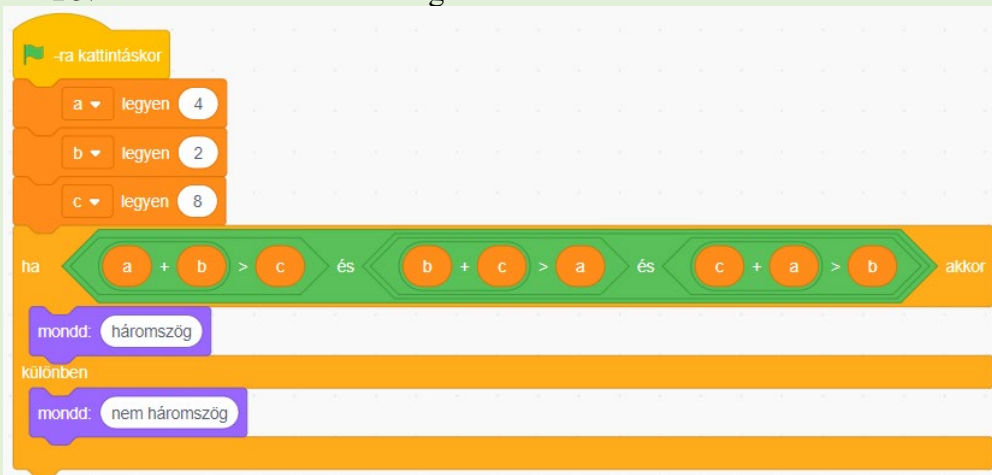


Mit figyelsz meg? Keresz más, pontosan 3 olyan utasítást, amelynek ugyanez a hatása, függetlenül  $x$  és  $y$  értékétől.

8. Írj algoritmust, amely beolvas az  $a$  és  $b$  változóba egy-egy nem nulla természetes számot és kiírja mennyivel kell csökkenteni  $b$  értékét, hogy az  $a$  kétszeresét kapjuk.

9. Emlékezz, mit jelent  $x / 10$  maradéka. Tudva, hogy  $x / 10$  lefelé kerekítve az  $x$ -nek 10-zel való osztási hányadosát jelenti, írd algoritmust, amely beolvas  $x$ -be egy pontosan háromjegyű természetes számot,  $x$ -ből kitörli a tízesek számjegyét és kiírja az eredményt. **Például:** ha  $x$  értéke 285 az eredmény 25.

10. Tekintsük a következő algoritmust:



- í. Fogalmazd meg az algoritmushoz tartozó feladatot!
- íí. Keres értékeket az  $a$ ,  $b$  és  $c$  változóknak úgy, hogy kiírja: **háromszög**.

11. Tekintsük a következő algoritmust:



```

- ra kattintáskor
a legyen 5
b legyen 5
c legyen 5
ha (a + b > c) és (b + c > a) és (c + a > b) akkor
ha (a = b) és (b = c) akkor
mond: Egyenlő oldalú háromszög
különben
ha (a = b) vagy (a = c) vagy (b = c) akkor
mond: Egyenlő szárú háromszög
különben
mond: Általános háromszög
különben
mond: Nem képez háromszöget!

```

- í. Fogalmazd meg az algoritmushoz tartozó feladatot!
- íí. Próbáld egyedül! Mi jelenik meg a képernyőn, ha  $a=4$ ,  $b=3$ ,  $c=6$ ?

12. Tekintsük a következő algoritmust:

```

- ra kattintáskor
mond: Kérem az "a" értékét 2 másodpercig
kérd: válasz és várj
a legyen válasz
s legyen 0
ismételd eddig: (a = 0)
s legyen (s + a / 10) maradéka
a legyen (a / 10) lefelé kerekítve
mond: s

```

Fogalmazd meg az algoritmushoz tartozó feladatot!

13. Tekintsük a következő algoritmust:

- í. Fogalmazd meg az algoritmushoz tartozó feladatot!
- íí. Módosítsd az algoritmust úgy, hogy kiírja:

```

-ra kattintáskor
mondj: Kérem az "a" értékét 2 másodpercig
kérdj: válasz és várj
a legyen válasz
mszj legyen 0
ismételd eddig: a = 0
ha a / 10 maradéka > mszj akkor
    mszj legyen a / 10 maradéka
a legyen a / 10 lefelé kerekítve
mondj: mszj

```

- a. az **a** legkisebb számjegyét;
- b. hányszor jelenik meg egy **c** (beolvasott) számjegy az **a**-ban;
- c. a legnagyobb számjegyet, és az előfordulási számát **a**-ban.

**14.** Tekintsük a következő algoritmust:

```

-ra kattintáskor
mondj: Kérem az "a" értékét 2 másodpercig
kérdj: válasz és várj
a legyen válasz
mondj: Kérem a "b" értékét 2 másodpercig
kérdj: válasz és várj
b legyen válasz
ismételd eddig: b = 0
    r legyen a / b maradéka
    a legyen b
    b legyen r
mondj: a

```

Fogalmazd meg az algoritmushoz tartozó feladatot!

**15.** Tekintsük a következő algoritmust

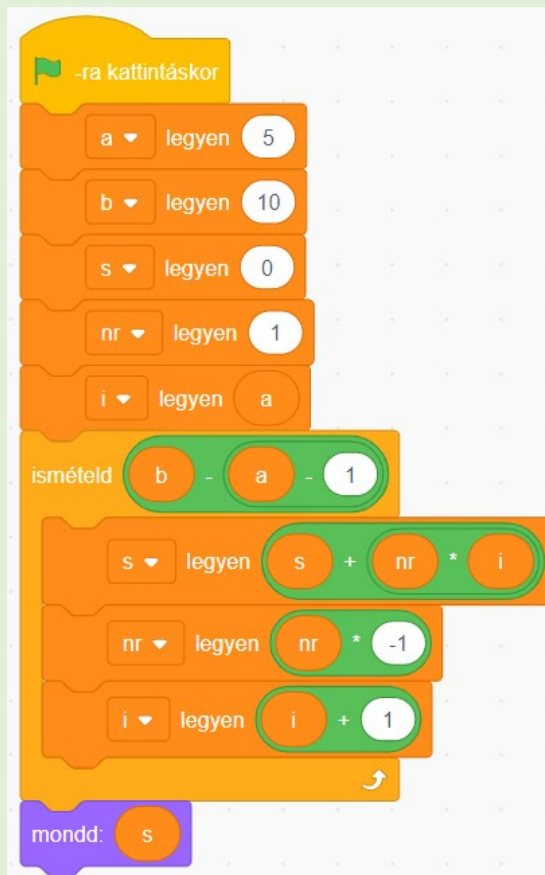
```

-ra kattintáskor
mondj: Kérem az "a" értékét 2 másodpercig
kérdj: válasz és várj
a legyen válasz
k legyen 0
ha a = 0 akkor
    k legyen 1
ismételd eddig: a = 0
    ha a / 10 maradéka = 0 akkor
        k legyen k + 1
    a legyen a / 10 lefelé kerekítve
mondj: k

```

- i. Fogalmazd meg az algoritmushoz tartozó feladatot!
- ii. Módosítsd az algoritmust úgy, hogy kiírja:
  - a. az **a** számjegyeinek számát;
  - b. az **a** páros számjegyeinek számát.

16. Tekintsük a következő algoritmust:



```
Scratch code for problem 16:
-ra kattintáskor
a legyen 5
b legyen 10
s legyen 0
nr legyen 1
i legyen a
ismételd b - a - 1
  s legyen s + nr * i
  nr legyen nr * -1
  i legyen i + 1
mondd: s
```

- í. Fogalmazd meg az algoritmushoz tartozó feladatot! Általánosíts!
- íí. Tudsz írni egy olyan algoritmust, amely nem használ egyetlen ismétlődő struktúrát sem?

17. Tekintsük a következő algoritmust:




```
Scratch code for problem 17:
-ra kattintáskor
a legyen 36
ismételd addig: a / 2 maradéka > 0
  a legyen a / 2 lefelé kerekítve
k legyen 0
i legyen 1
ismételd addig: i > a / 2 lefelé kerekítve
  ha a / i maradéka = 0 akkor
    k legyen k + 1
  i legyen i + 2
k legyen k + 1
mondd: k
```

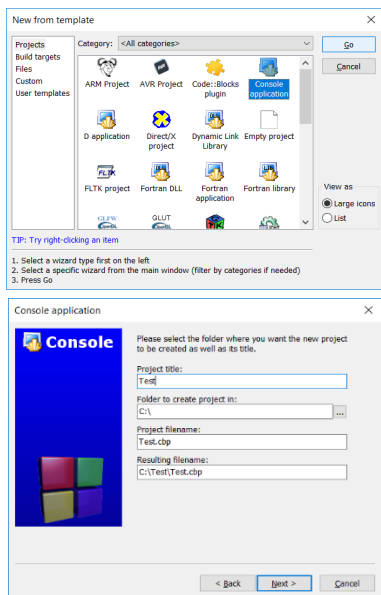
- í. Fogalmazd meg az algoritmushoz tartozó feladatot!
- íí. Módosítsd az algoritmust, úgy hogy kiírja:
  - a. az **a** páratlan osztóinak összegét;
  - b. az **a** minden osztójának összegét;
  - c. azt, hogy **a** prímszám-e;
  - d. az **a** páros osztóinak számát.



## EGY PROGRAMOZÁSI NYELV FEJLESZTŐI KÖRNYEZETÉNEK LEHETŐSÉGEI: SZERKESZTÉS, FUTTATÁS, HIBAKERESÉS

Az egyik leggyakrabban használt programozási környezet a CodeBlocks. Ebben a C/C++ programozási nyelven lehet kódolni. A programcsomag telepítése a letöltése után történik. Az egyik hely, ahonnan le lehet tölteni az OJI Kit nevű programcsomagot, ami tartalmazza a CodeBlocks-ot is: <http://olimpiada.info/oji2019/index.php?cid=regulament>.

A telepítés után, a munkaasztalon meg fogod találni a CodeBlocks indító ikonját . Ajánlott, hogy mielőtt elkezded egy programkód írását, hozz létre egy új projektet. Ehhez a következő lépéseket kell megtenni:

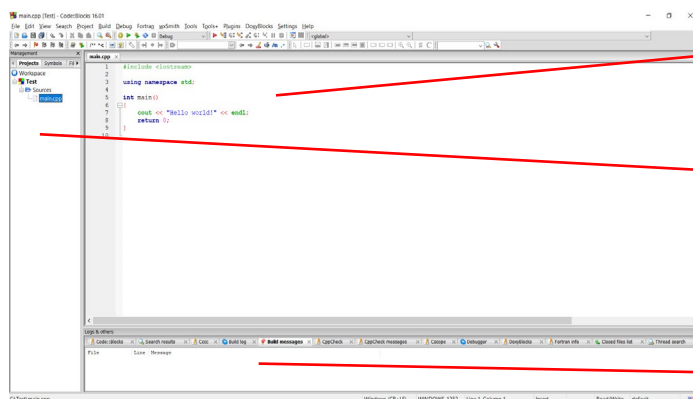


1. File -> New -> Project

2. Válaszd ki „Console application”-t majd kattints a „Go” gombra. A következő ablakból a „Next”-re kell kattintani. Meg fog jelenni egy ablak, ahonnan ki kell választod, hogy milyen programozási nyelvet fogsz használni: C vagy C++. Mivel ebben a tankönyvben a C++ nyelvet mutatjuk be, ajánlott, hogy azt válassz ki.

3. A következő lépésben kell megadni a projekt nevét és azt a helyet, ahová elmented. Például, a mellékelt képen, a projekt neve „Test” és a C:\ lemezegegyesre lesz mentve. A név és a mentési hely megadása után a „Next” gombra kell kattintani. A következő ablakban ne módosíts semmit. Csak kattints a „Finish” gombra.

A CodeBlocks felhasználói felületének fontosabb elemei:



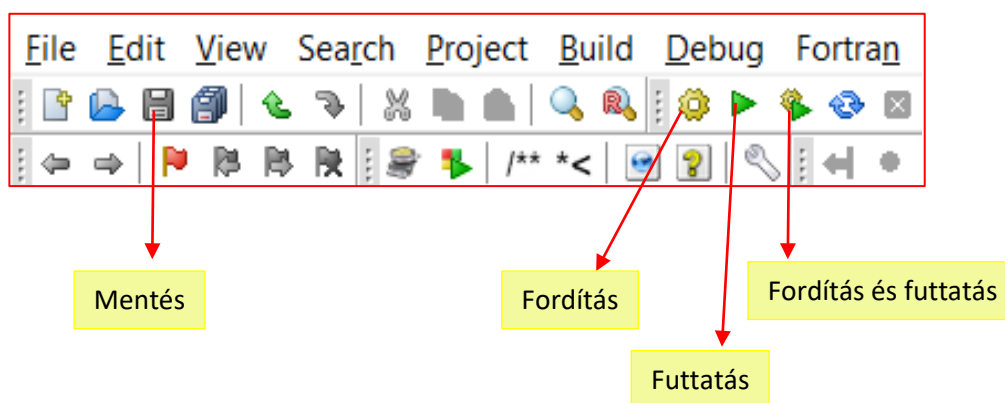
Ebbe a részbe szerkesztetted a programodat.

Itt jelenik meg a projekthez tartozó minden állomány. Ezek közül a leggyakrabban használt a main.cpp.

Ebben az ablakban jelennek meg esetenként a programodban elkövetett hibák és az alkalmazás üzenetei.

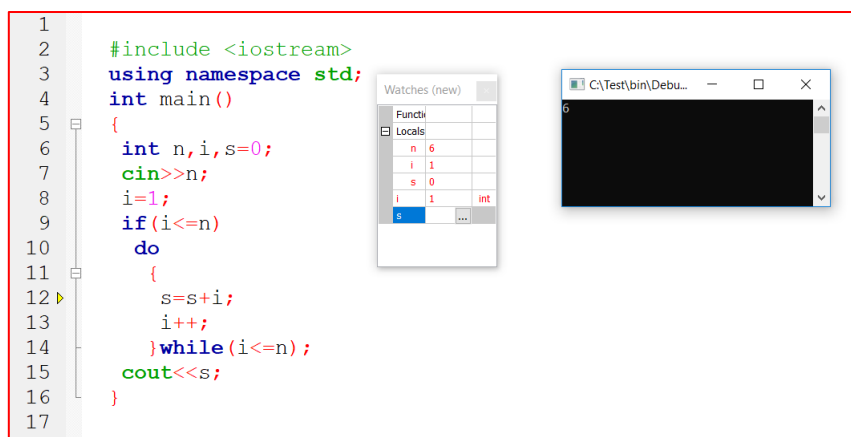
Egy alkalmazás kidolgozásának lépései:

1. **Forráskód szerkesztése** – a program tényleges írását jelenti.
2. **Fordítás** – az általa C++ nyelven írt programkódot tartalmazó fájlok (forrásfájlok) alapján egy fordító (compiler) létrehozza a gépi kódú utasításokat tartalmazó fájlokat. Ez nem történik meg, ha a program szintaktikai hibákat tartalmaz.
3. **Futtatás** – a lefordított program futtatása, végrehajtása.
4. **Tesztelés** – ezt a lépést a felhasználó végzi. Különböző bemeneti adatok esetén vizsgálja, hogy a kimeneti adatok megfelelnek-e az elvárt értékeknek.
5. **Hibakeresés** – ha hibák merülnek fel a tesztelés során, ezeket kijavítva a 2. lépéstől folytatjuk.



A hibakeresésnek (Debug) két fő része van, amelyek segítségével meg tudjuk „javítani” a programokat, ahogy a következő képen is látszik:

- a. Debug → Debugging windows → Watches, lehetővé teszi a változók értékének ismeretét minden időpillanatban.
- b. Debug → Step into (vagy Shift+F7) és aztán F7, segíti követni a program működését sorról-sorra



Elkezdheted megszerkeszteni a programkódodat a **main.cpp** állományba. Ez a projekt legfontosabb állománya. Itt a programod a felhasználó számára érthető formában található.



Feladatok

1. Betűzd (a-e) meg az alkalmazás kidolgozásának lépéseit, a sorrendnek megfelelően.
  - ..... futtatás
  - ..... tesztelés
  - ..... forráskód szerkesztés
  - ..... hibakeresés
  - ..... fordítás
  
2. Egészítsd ki a következő mondatokat:
  - a. Ajánlott egy ..... létrehozása a program effektív írása előtt.
  - b. Egy program hibakeresésére az alkalmazás menüjéből a..... használjuk.



## OPERÁTOROK

Amikor egy idegen nyelvet kezdtél tanulni először egyszerű szavakat tanultál: főneveket, igéket és kötőszavakat, csak később alkottál ezek segítségével egyszerű és bővített mondatokat. A programozás tanulása során is hasonló a helyzet, előbb az úgynevezett operátorokkal kell megismerkedned.

Az operátoroknak három nagy csoportját különböztetjük meg:

### 1. Aritmetikai operátorok

A legtöbb ilyen operátort ismered már matematikából:

+ (összeadás), - (kivonás), \* (szorzás)

Két új operátort ismerhetsz meg a programozás tanulása során:

/ – két szám osztását jelenti, amennyiben az egyik szám valós típusú, illetve az egész osztás hányadosát, ha mindkét szám egész típusú;

% – csak akkor használható, ha mindkét szám egész, eredményül pedig az osztás maradékát adja;

#### *Példák:*

$7/3 = 2$  (mivel mindkét szám egész, az eredmény az osztás hányadosa lesz);

$6.4/2=3.2$  (mivel az első szám valós, a művelet az osztás eredménye lesz);

$7\%3=1$  (7-nek 3-mal való osztási maradéka);

$7\%2.4$  NEM LEHET ELVÉGEZNI (a második szám nem egész típusú).

#### *Sajátos esetek*

$x/10$  – letörli az  $x$  egész szám utolsó számjegyét;

$x\%2$  – az eredmény  $0$ ,  $1$  vagy  $-1$ . Ha  $x$  páros szám, akkor az eredmény  $0$ , ha  $x$  páratlan és pozitív, akkor  $1$ , ha pedig  $x$  páratlan negatív szám akkor  $-1$ . Vagyis az  $x\%2$  segítségével az  $x$  egész szám paritását vizsgáljuk;

$x\%10$  – az  $x$  egész szám utolsó számjegyével egyenlő.

### 2. Relációs operátorok

Az operátorok ezen típusai a  $0$  vagy  $1$  eredményt szolgáltatják. Megjegyezzük, hogy az informatikában a  $0$  a hamis, az  $1$  pedig az igaz logikai értéknek felel meg.

<, <=, >, >= operátorokat a matematikai értelemben használjuk;

== abban az esetben használjuk, ha ellenőrizni szeretnénk, hogy a bal és jobb oldali kifejezések értékei megegyeznek-e;

!= abban az esetben használjuk, ha ellenőrizni szeretnénk, hogy a bal és jobb oldalon lévő kifejezések értékei különbözőek-e.

#### *Példa*

$7==8$  – értéke  $0$

$5==5$  – értéke  $1$

$7!=8$  – értéke  $1$

$5!=5$  – értéke  $0$

### 3. Logikai operátorok

&& - és (and) || - vagy (or) ! -tagadás (not)

A logikai operátorok használata:

1.kifejezés && 2.kifejezés – akkor igaz, ha mindkét kifejezés egyszerre igaz;  
 1.kifejezés || 2.kifejezés – akkor igaz, ha legalább az egyik kifejezés igaz;  
 ! kifejezés – akkor igaz, ha a kifejezés hamis.

**Példák:**

➤ Ha a következő kifejezést akarod leírni informatikai operátorokkal  $x \in [a, b]$ , mivel nem létezik az  $\in$  operátor, a következő formát használjuk:

$$(x \geq a) \ \&\& \ (x \leq b)$$

➤ Hasonlóan az  $x \notin [a, b]$ , esetén

$$(x < a) \ || \ (x > b)$$

vagy

$$! \ ( (x \geq a) \ \&\& \ (x \leq b) )$$



1. Feleltess meg a bal oldali oszlop minden elemének egy értéket a jobb oldali oszlopból:

7/11	-2
12%5	0
7+2	6
5-7	2
17%10	9
2*3	7

2. Számítsd ki a következő kifejezéseknek megfelelő értékeket:

- $((7+2) \% 5 + (9-3) / 3) / 1.5$
- $27 \% 10 + 27 / 10$
- $6 \% 10 + 273 / 10 \% 10 + 273 \% 100$

3. Adj példát olyan kifejezésekre, amelyek a / és % operátorokat használják és elvégzik a következő számításokat:

- kitörli a **845** szám tízes helyiértéken lévő számjegyét;
- meghatározza az **578** szám fordítottját (tükörképét)

## EGY PROGRAM SZERKEZETE

Ez előző osztályokban tanultak során észrevehetted, hogy az algoritmusoknak sajátos szerkezete van, hasonlóan a C++ nyelvben megírt programokéhoz:

1. Fejállományok beillesztése;
2. Változók deklarálása;
3. Bemeneti adatok beolvasása;
4. A program sajátos utasításai;
5. Kimeneti adatok kiírása.

A következőkben részletesen megismerkedünk a program fent említett egységeivel.

A negyedik alponthan említett *sajátos utasítások*, azon utasítások összessége, amelyek feldolgozzák a bemeneti adatokat az eredmény meghatározása céljából. Ezeket az utasításokat a megoldandó feladat határozza meg.

### 4.4.1. A programozási nyelv szókészlete

Fejállományok beillesztése

A fejállományok előre értelmezett függvényeket tartalmaznak. Például a hatványok kiszámítása történhet ismételt szorzások segítségével vagy használva a `pow(a, b)` függvényt, amely a `cmath` fejállományban található.

A leggyakrabban használt fejállományok:

`iostream` – tartalmazza a billentyűzetről való beolvasásra használt függvényeket (`cin`>>) és a képernyőre íratásra használt függvényeket (`cout`<<).

`cmath` – matematikai függvényeket tartalmaz. Ezek közül a leggyakrabban használjuk: `pow(a, b)`, amely kiszámítja  $a^b$  értékét valamint az `sqrt(a)`, amely  $\sqrt{a}$ -t határozza meg.

A fejállomány beillesztésének módja a programba:

```
#include <fejállomany_neve>
```

Például:

```
#include <iostream>
```

Amennyiben több állomány beillesztésére van szükségünk ezeket a program elejére, az első sorokba írjuk. Ha egy billentyűzetről beolvasott természetes számról szeretnénk eldönteni, hogy teljes négyzet-e, be kell olvasnunk a számot, kiszámítanunk a négyzetgyökét és ki kell íratnunk a megfelelő üzenetet. Tehát szükségünk van mindkét fejállományra:

```
#include <iostream>
```

```
#include <cmath>
```



## 4.4.2. Numerikus (szám típusú) adatok

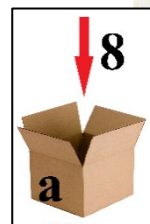
A grafikus elemekkel ellentétben a C++ programozási nyelvben használat előtt minden változót deklarálni kell.

A C++ nyelvben a változók egy memóriarészt foglalnak le, van egy nevük és egy általuk tartalmazott érték.

Például `int a=8`; egy egész típusú számot jelöl, amelynek neve `a` és értéke `8`.

A leggyakrabban használt adattípusok a C++ nyelvben:

- a) `int` vagy `long` – pozitív és negatív számok, amelyek maximum 9 számjegyből állnak;
- b) `long long` – legtöbb 18 számjegű pozitív és negatív számok;
- c) `float` – pozitív és negatív valós számok.



**FIGYELEM!**

Nagy különbség van a következő két deklaráció között: `int a=7`; és `float a=7`;. Az első esetben az `a` változó a 7-es egész számot őrzi, a második esetben a `7.00000` valós számot!

**Megjegyzés:** Egy változó átalakítható egy másik típusra a következőképpen: *(típus) változó*.

Például:

A következőket írhatja ki:

```
int a=12,b=5;           2
cout<<a/b<<endl;       2.4
cout<<a/(float)b;
```

## 4.4.3. Adatok beolvasása és kiírása

A programok többsége bemeneti adatokból indulva és őket feldolgozva jut el a kimeneti adatokhoz. Általában ezeket az adatokat a billentyűzetről olvassuk be a fent említett `cin>>` utasítás segítségével.

Amennyiben két változó értékét szeretnénk beolvasni, a következő utasításokat használhatjuk:

```
cin>>a>>b; vagy cin>>a;
cin>>b;
```

A kimeneti adatokat a képernyőre a `cout<<` utasítás segítségével írhatjuk

**1. példa:**

```
int a=8, b=17;
cout<<a<<b;
```

Kiírja: `817` (egybe írja a számokat, így nem tudjuk pontosan, hogy melyik a két szám)

## 2. példa

```
int a=8, b=17;  
cout<<a<<" "<<b;
```

Kiírja: 8 17 (a számokat szóközzel választjuk el)

## 3. példa

```
int a=8, b=17;  
cout<<a<<'\n'<<b;
```

Kiírja: 8 (a számokat külön sorokba írja)

17

Ha szöveget íratunk a képernyőre, akkor az " " jeleket használjuk.

```
cout<<"Informatika és IKT óra";
```

Kiírja a képernyőre a következő szöveget: *Informatika és IKT óra*

### Feladatok

1. Az alábbiak közül melyik olvas be két legtöbb 9 számjegyű egész számot?

a. `int a;`  
`float c;`  
`cin>>a>>b>>c;`

b. `int a,b;`  
`cin>>b;`

c. `float a,b;`  
`cin>>a>>b;`

d. `int a,b;`  
`cin>>a>>b;`

2. A következő változatok közül melyik írta ki:  $1+7=8$ ?

a. `int a=1,b=7;`  
`cout<<"a+b=8";`

b. `int a=1,b=7;`  
`cout<<a<<"+"<<b<<"=a+b";`

c. `int a=1,b=7;`  
`cout<<a<<"+"<<b<<"="<<a+b;`

d. `int a=1,b=7;`  
`cout<<"1+7="<<a*b;`

3. A jobb oldali oszlopban lévő változóknak feleltess meg a helyes típust a bal oldali oszlopból.

a. `int`

1. `a=3.475`

b. `float`

2. `c=4356`

c. `long long`

3. `b=-35627487896`

#### 4.4.4. Lineáris szerkezet programozási nyelvbe való átültetésére vonatkozó utasítások

Ahogy már eddig észrevehetted minden parancsot a ; jellel zárunk le. Az informatikában a parancsokat utasításoknak nevezzük.

**Példa:**

Írj programot, amely beolvas két **a** és **b** legtöbb 5 számjegyet tartalmazó egész számot, melyek egy téglalap szélességét és hosszúságát jelentik. A program írja ki a téglalap kerületét és területét.

**Megoldás:**

```
#include <iostream>
using namespace std;
int main()
{
    int a,b,T,K;
    cout<<"Ird be a teglalap hosszusat: ";
    cin>>a;
    cout<<"Ird be a teglalap szelesseget: ";
    cin>>b;
    T=a*b;
    K=2*(a+b);
    cout<<"Terulet= "<<T<<endl;
    cout<<"Kerulet= "<<K;
}
```

A következő jelenik meg a program futtatásakor:

```
Ird be a teglalap hosszusat: 5
Ird be a teglalap szelesseget: 6
Terulet= 30
Kerulet= 22
Process returned 0 (0x0)   execution time : 2.468 s
Press any key to continue.
```

Feladatok

1. Írd meg az alábbival egyenértékű C++ programot:

```
-ra kattintáskor
kérdézd Kérem az "a" értékét és várj
a legyen válasz
kérdézd Kérem a "b" értékét és várj
b legyen válasz
t legyen a * b
z legyen a + b
n legyen t * 2 + 3 * z
c legyen n / 10 maradeka + n / 10 lefelé kerekítve
mondj n és és c összefűzve összefűzve
```



2. Olvass be egy pontosan háromjegyű  $n$  természetes számot és írasd ki az  $n$  szám számjegyeinek összegét!

**Példa**

Beolvassuk: 254

Kiírja: 11

3. Olvass be egy legtöbb négyjegyű  $n$  természetes számot, majd számítsd ki a következő összeget:

$$1+2+3+\dots+n$$

**Példa**

Beolvassuk: 7

Kiírja: 28

4. Bukarest és Ploiești városok közötti távolság 60 de km. Írj egy olyan C++ programot, amely beolvas a billentyűzetről egy szigorúan pozitív  $v$  számot, amely egy vonat sebességét jelenti km/h-ban kifejezve, majd írasd ki, hogy hány perc alatt érkezik meg Ploieștire a Bukarestből induló vonat (idő=távolság/sebesség)!

**Példa**

Beolvassuk: 50

Kiírja: 72

5. Írj C++ programot, amely beolvassa az  $x$  természetes számot, amely egy film hosszát jelenti percben kifejezve. Írasd ki, hogy hány órát és hány percet tart a film!

**Példa**

Beolvassuk: 200

Kiírja: 3 óra 20 perc

6. Írj olyan programot, amely beolvas három  $a$ ,  $b$  és  $c$  természetes számokat, majd kiírja a három szám számtani középarányosát!

**Példa:**

Beolvassuk: 5 6 10

Kiírja: 7

7. Jánosnak van  $x$  darab kecskéje,  $y$  darab tyúkja és  $z$  darab juha. Írj programot, amely beolvassa az  $x$ ,  $y$  és  $z$  számokat, majd kiírja, hogy János állatainak összesen hány lába van!

**Példa:**

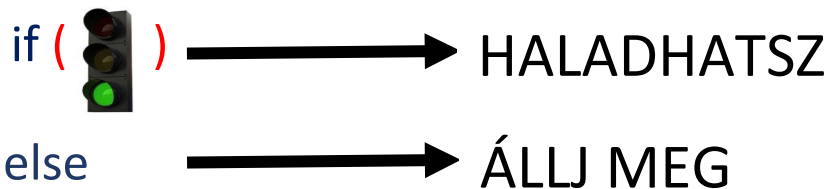
Beolvassuk: 3 2 4

Kiírja: 32

## AZ ELÁGAZÁSI SZERKEZET IMPLEMENTÁLÁSA PROGRAMOZÁSI NYELVBEN

Az eddigi programok utasítások sorozatát tartalmazták, de nem használtunk elágazásokat, amelyek egy adott feltétel értelmezése után bizonyos utasításokat hajtanak végre, ha igaz a feltétel és más utasításokat, ha a feltétel hamis. A grafikus blokkok esetén az elágazás a *ha* struktúra volt. A C++ programozási nyelvben erre az **if** utasítást használjuk.

Ez az utasítás az utcai jelzőlámpához hasonlóan működik: ha a zöld szín világít, át lehet haladni, ha pedig nem zöld a színe, akkor meg kell állnunk.



Az **if** utasítás általános alakja:

```
if (feltétel)
{
    1. utasítás sorozat
}
else
{
    2. utasítás sorozat
}
```

**Megjegyzések:**

- Amennyiben az utasítás sorozatok csak egyetlen utasításból állnak, akkor a kapcsos zárójelek (**{ }**) elhagyhatók.
- Az **if** utasítás használható az **else** ág nélkül is.

**Példa:**

Beolvasunk két egész számot és kiíratjuk a kettő közül a nagyobbat. A feladat többféleképpen is megoldható:

1. változat

```
#include <iostream>
using namespace std;
int main()
{
    int a,b,maximum;
    cout<<"Kerem az a erteket: ";
    cin>>a;
    cout<<"Kerem a b erteket: ";
    cin>>b;
    if(a<b)
        maximum=b;
    else
        maximum=a;
    cout<<"A legnagyobb ertek: "<<maximum;
}
```

2. változat

```
#include <iostream>
using namespace std;
int main()
{
    int a,b,maximum;
    cout<<"Kerem az a erteket: ";
    cin>>a;
    cout<<"Kerem a b erteket: ";
    cin>>b;
    maximum=a;
    if(a<b)
        maximum=b;
    cout<<"A legnagyobb ertek: "<<maximum;
}
```

- A fenti példát két változatban is megoldottuk, hogy felhívjuk a figyelmeteket a következőkre:
- Egy feladat sokféleképpen megoldható, nem csak egyetlen elfogadható változat létezik.
  - Észrevehetjük, hogy a második megoldás során az `if` utasítás nem tartalmazott `else` ágat.

## Feladatok

1. Az alábbi változatok közül melyik írja ki helyesen két  $a$  és  $b$  szám különbségének abszolút értékét?

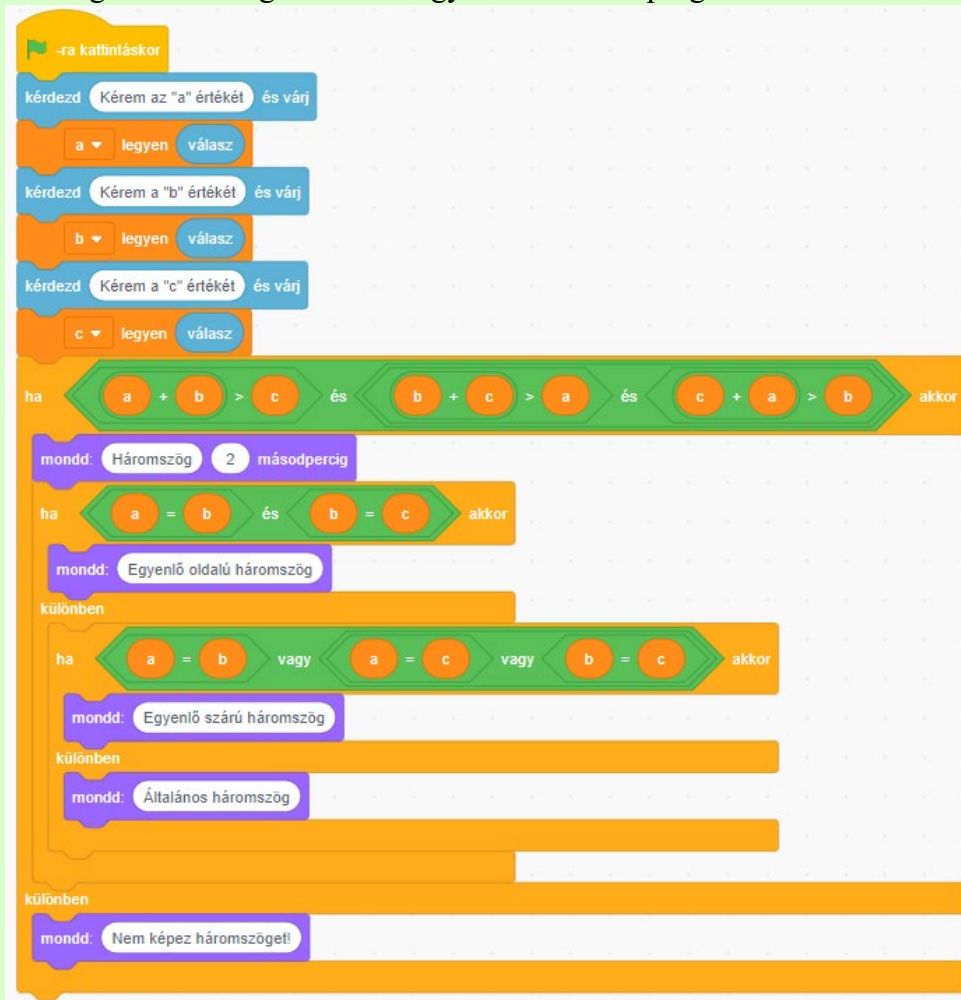
a. `if (a < b) cout << a - b;`  
`else cout << b - a;`

b. `if (a < b) cout << (a + b) / 2;`  
`else cout << b - a;`

c. `if (a > b) cout << a - b;`  
`else cout << b - a;`

d. `if (a < b) cout << a - b / 2;`  
`else cout << b / 2 - a;`

2. Írd meg az alábbi algoritmussal egyenértékű C++ programot!





3. Mit írat ki a mellékelt program az alábbi két esetben?

a. a=3, b=8

b. a=7, b=3

```
#include <iostream>
using namespace std;
int main()
{
    int a,b,c;
    cout<<"Kerem az a erteket: ";
    cin>>a;
    cout<<"Kerem a b erteket: ";
    cin>>b;
    if(a>b)
    {
        c=a;
        a=b;
        b=c;
    }
    cout<<a<<" "<<b;
}
```

4. Beolvasunk három **a**, **b** és **c** egész számot. Írj olyan C++ programot, amely kiírja a „**lehetséges**” üzenetet, ha az egyik szám felírható a másik kettő összegeként és a „**nem lehetséges**” üzenetet ellenkező esetben!

**Példák**

Beolvassuk: 3 8 5

Kiírja: **Lehetséges!**

Beolvassuk: 2 9 4

Kiírja: **Nem lehetséges!**

5. Beolvasunk egy **n** egész számot. Írj programot, amely kiírja a „**páros szám**” üzenetet, ha a beolvasott szám páros és a „**páratlan szám**” üzenetet, ha a szám páratlan!

**Példák**

Beolvassuk: 4

Kiírja: **Páros szám!**

Beolvassuk: 7

Kiírja: **Páratlan szám!**

6. Olvasd be az **n** valós számot és annak függvényében, hogy az **n** szám szigorúan pozitív, szigorúan negatív, vagy nullával egyenlő íráss ki megfelelő üzenetet a programban!

**Példák**

Beolvassuk: 5

Kiírja: **Pozitív szám!**

Beolvassuk: -2

Kiírja: **Negatív szám!**

Beolvassuk: 0

Kiírja: **A szám egyenlő nullával!**

## ISMÉTLŐ SZERKEZET IMPLEMENTÁLÁSA PROGRAMOZÁSI NYELVBEN

Előfordult már veled az előző algoritmusok során, hogy bizonyos utasításokat többször szeretnél volna elvégezni? Több azonos sor beírása helyett használhatjuk az úgynevezett ismétlő utasításokat.

Az ismétlő utasítás tehát többször végrehajt egy utasítás sorozatot, annak függvényében, hogy teljesül-e egy adott feltétel.

Például: Menj előre addig, amíg az út egyenesen halad.

A C++ programozási nyelvben három fajtája létezik az ismétlő utasításoknak, amelyek egyenértékűek a grafikus blokkoknál használt ismétlésekkel.

### C++

**while**

**for**

**do... while**

### Grafikus blokkok

**ismételd eddig:**

### 4.6.1. A while ismétlő utasítás

Elöltesztelő ismétlésnek is nevezzük, mivel a feltétel kiértékelése az utasítás sorozat végrehajtása előtt történik.

A **while** utasítás általános alakja:

```
while (feltétel)
{
    1. utasítás;
    2. utasítás;
    .....
}
```

#### Megjegyzés:

Az ismétlések (ciklusok) esetén mindig úgy módosítsd a változók értékeit, hogy a feltétel, amelyet folyamatosan kiértékel, bizonyos számú lépés után hamissá válhasson, mert csak így állhat meg az ismétlés. Amennyiben a feltétel mindig igaz, tehát nem tud megállni az ismétlés, azt mondjuk, hogy „végtelen ciklushoz” jutottunk.

A mellékelt program egyszer sem fogja kiírni az **i** változó értékét, ha egy nullától különböző számot olvasunk be, mivel a **while** utasítás feltételének logikai értéke soha nem lesz **0**, vagyis **hamis**.

```
#include <iostream>
using namespace std;
int main()
{
    int a,i;
    cin>>a;
    i=1;
    while(a!=0)
        i=i+1;
    cout<<i;
}
```

**Példák:**

<p>1. Beolvasunk egy nullától különböző természetes <b>a</b> számot. Írj programot, amely növekvő sorrendben írhatja ki az <b>a</b>-nál kisebb, vagy egyenlő nullától különböző természetes számokat!</p>	<p>2. Írj programot, amely beolvas egy <b>n</b> nem nulla természetes számot és kiírja az <b>n</b> számjegyeinek a számát!</p>
<pre>#include &lt;iostream&gt; using namespace std; int main() {     int a,i;     cin&gt;&gt;a;     i=1;     while(i&lt;=a)     {         cout&lt;&lt;i&lt;&lt;" ";         i=i+1;     } }</pre>	<pre>#include &lt;iostream&gt; using namespace std; int main() {     int n,szj=0;     cout&lt;&lt;"Kérem az n értéket: ";     cin&gt;&gt;n;     while(n!=0)     {         szj=szj+1;         n=n/10;     }     cout&lt;&lt;"A számjegyek száma: "&lt;&lt;szj; }</pre>



1. Az alábbi utasítás sorozatok közül melyik fogja kiírni növekvő sorrendben az **n**-nél kisebb vagy egyenlő, nullától különböző, páros természetes számokat?

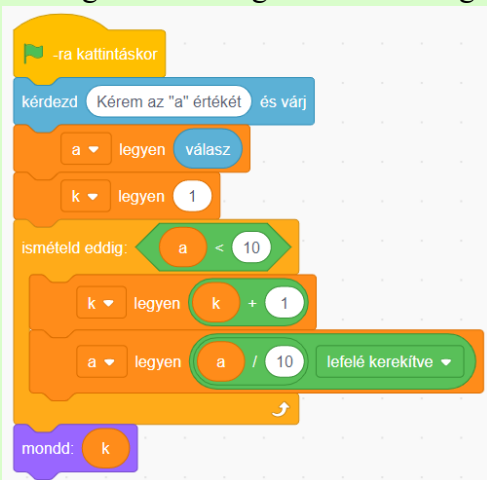
a. `i=2;`  
`while(i<=n)`  
`{`  
 `cout<<i<<" ";`  
 `i=i+2;`  
`}`

b. `i=1;`  
`while(i<=n)`  
`{`  
 `cout<<i<<" ";`  
 `i=i+2;`  
`}`

c. `i=1;`  
`while(i<=n)`  
`{`  
 `if(i%2!=0)`  
 `cout<<i<<" ";`  
 `i=i+1;`  
`}`

d. `i=0;`  
`while(i<=n)`  
`{`  
 `if(i%2==0)`  
 `cout<<i<<" ";`  
 `i=i+2;`  
`}`

2. Írd meg az alábbi algoritmusnak megfelelő C++ programot!





3. Beolvassunk egy  $a$  ( $a < 10^9$ ) nullától különböző egész számot. Írj C++ programot, amely kiírja az  $a$  páros számjegyeinek a számát, ha nincs páros számjegye, akkor írjon ki egy megfelelő üzenetet!

**Példák**

Beolvassuk: 45792041

Kiírja: 4

Beolvassuk: 7139

Kiírja: A számnak nincs páros számjegye!

4. Beolvassunk egy  $n$  ( $n < 10^6$ ) természetes számot. Írj C++ programot, amely kiírja növekvő sorrendben az  $n$  természetes osztóit!

**Példa**

Beolvassuk: 12

Kiírja: 1 2 3 4 6 12

5. Írj C++ programot, amely beolvasson egy  $n$  ( $n < 10^6$ ) nullától különböző természetes számot és kiszámítja azt a legnagyobb  $2^k$  alakú számot, amely kisebb vagy egyenlő, mint  $n$ .

**Példák**

Beolvassuk: 9

Kiírja: 8

Beolvassuk: 1

Kiírja: 1

6. Beolvassunk egy  $n$  ( $n < 10^6$ ) természetes számot. Írj olyan C++ programot, amely megvizsgálja, hogy ez a szám palindrom-e, majd kiír egy megfelelő üzenetet. Egy számot, akkor nevezünk palindromnak, ha egyenlő a fordítottjával (tükörképével).

**Példák**

Beolvassuk: 19391

Kiírja: A szám palindrom!

Beolvassuk: 125251

Kiírja: A szám nem palindrom!

7. Beolvassunk két nullától különböző  $a$  és  $b$  ( $a < 10^6$  és  $b < 10^6$ ) természetes számot. Írj olyan C++ programot, amely meghatározza és kiírja a két szám legnagyobb közös osztóját!

**Példák**

Beolvassuk: 9 12

Kiírja: 3

### 4.6.2. A for ismétlő utasítás

Ezt az ismétlő utasítást, általában olyankor használjuk, amikor pontosan tudjuk, hogy hányszor kell megismételni az utasítás sorozatot. Rögzített lépésszámú ismétlésnek is nevezzük.

A **for** utasítás három kifejezést tartalmaz, ahogyan az alábbi példákban látható:

a. **for (i=a; i<=b; i=i+c)** – az **i** változó kezdőértéke egyenlő az **a** értékével és **c**-vel növekszik, amíg el nem éri a **b**-ben tárolt értéket. Abban az esetben használjuk, ha **a<=b**.

b. `for(i=a;i>=b;i=i-c)` az `i` változó kezdőértéke `a`-val egyenlő és minden lépésben csökken a `c` változó értékével, amíg kisebb nem lesz, mint `b`. Akkor használjuk, amikor `a>=b`.

A C++ nyelvben még használunk két fontos operátort, amelyek nagyon gyakran megjelennek a `for` ciklusokban:

- `++` → inkrementálás, vagyis növeli a változó értékét 1-gyel.
- `--` → dekrementálás, vagyis csökkenti a változó értékét 1-gyel.

A következők egyenértékűek:

- `for(i=a;i<=b;i=i+1)` írható így is: `for(i=a;i<=b;i++)`
- `for(i=a;i>=b;i=i-1)` írható így is: `for(i=a;i>=b;i--)`

### Példa:

1. Beolvasunk egy `n` ( $1 < n < 10^6$ ) természetes számot. Azt a C++ programot, amely megvizsgálja, hogy az `n` szám prím-e és kiírja a megfelelő üzenetet, megírhatjuk az alábbi módon:

```
#include <iostream>
using namespace std;
int main()
{
    int n,i,c=0;
    cout<<"Kerem az n erteket: ";
    cin>>n;
    for(i=2;i<n;i++)
        if(n%i==0)
            c++;
    if(c==0)
        cout<<"Primszam";
    else
        cout<<"Nem primszam";
}
```

Módosítsd a programot úgy, hogy helyes eredményt írjon ki minden nullától különböző `n`,  $n < 10^6$  természetes szám esetén;

### Megjegyzés

Hatékonyabban működik a program, ha az `i*i<=n` feltételt használjuk az `i<n` helyett.

2. Beolvasunk egy nullától különböző `n` természetes számot,  $n < 10^6$ . Az alábbi C++ program, kiírja növekvő sorrendben az `n`-nél kisebb vagy egyenlő páratlan természetes számokat:

```
#include <iostream>
using namespace std;
int main()
{
    int n,i;
    cout<<"Kerem az n erteket: ";
    cin>>n;
    for(i=1;i<=n;i=i+2)
        cout<<i<<" ";
}
```

Változtass a programon úgy, hogy az `n`-nél kisebb vagy egyenlő páros számok legyenek kiírva csökkenő sorrendben.

1. Az alábbiak közül melyik kódrészlet írja ki az összes olyan, 0-tól különböző természetes számokat, amelyek kisebbek, mint  $n$ , növekvő sorrendben?

a. `for(i=1;i<=n;i--)`  
`cout<<i<<" "`;

b. `for(i=0;i<=n;i++)`  
`cout<<i<<" "`;

c. `for(i=n;i>=1;i--)`  
`cout<<i<<" "`;

d. `for(i=1;i<=n;i++)`  
`cout<<i<<" "`;

2. Írd át C++ nyelvre a következő algoritmust, `for` ciklust használva az `ismételd` helyett!



3. Írj C++ programot, amely beolvassa az  $n$  ( $n < 10^6$ ) természetes számot, majd ellenőrzi, hogy  $n$  tökéletes szám, és megfelelő üzenetet ír ki erre vonatkozóan. Tökéletes az a szám, amelynek kétszerese egyenlő az osztóinak összegével. **Például:**

Ha a beolvasott szám: 6

a kiírt üzenet: **Ez tökéletes szám.**

Ha a beolvasott szám: 8

a kiírt üzenet: **Ez nem tökéletes szám.**



4. Írj C++ programot, amely beolvassa az  $n$  ( $n < 10^6$ ) természetes számot és kiírja az összes olyan prímszámot, amely kisebb, mint  $n$ , csökkenő sorrendben.

**Példa:**

A beolvasott érték: 10

A kiírt értékek: 7 5 3 2

5. Írj C++ programot, amely beolvassa az  $n$  ( $n < 10^6$ ) természetes számot, és kiírja az összes olyan teljes négyzetet, amely kisebb, mint  $n$ , csökkenő sorrendben.

**Példa:**

Bemenet: 17

Kimenet: 16 9 4 1

### 4.6.1. A do ... while ismétlő utasítás

A do ... while ismétlő utasítást *hátultesztelő*-nek nevezik, mivel a feltétel teljesülésének ellenőrzése a ciklusmag végrehajtása **után** történik:

```
do
{
    utasítás1;
    utasítás2;
    .....
}while(feltétel);
```

**FIGYELEM!**

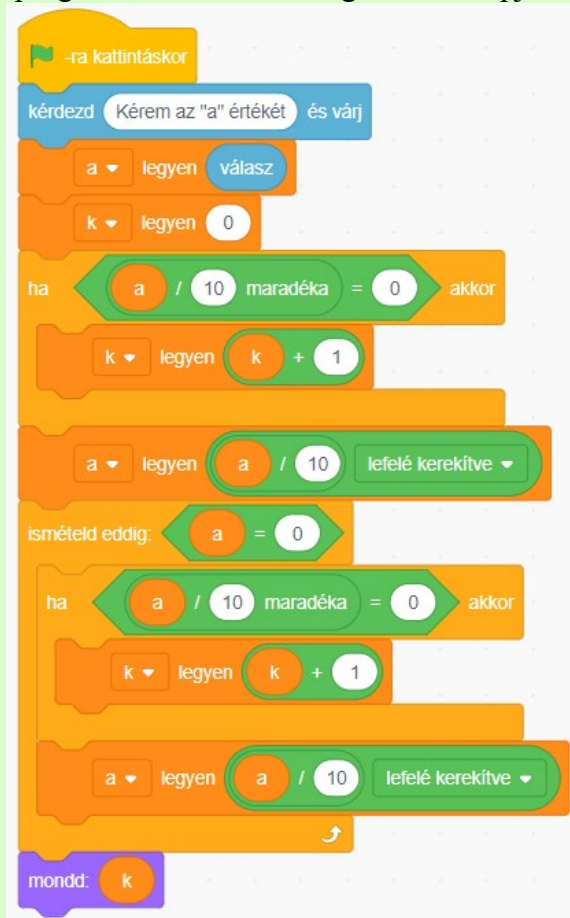
A do ... while az egyetlen olyan utasítás, amelyben a **(feltétel)** után ; van.

**Példa:**

Beolvassuk az  $n$  ( $n < 10^6$ ) természetes számot. A következő C++ program kiírja az  $n$ , szám számjegyeinek számát:

```
#include <iostream>
using namespace std;
int main()
{
    int n, nr=0;
    cout<<"Kerem az n erteket: ";
    cin>>n;
    do
    {
        nr++;
        n=n/10;
    }while(n!=0);
    cout<<"A szamjegyek szama: "<<nr;
}
```

Írj C++ programot a következő algoritmus alapján, de **do... while** ciklust használva:

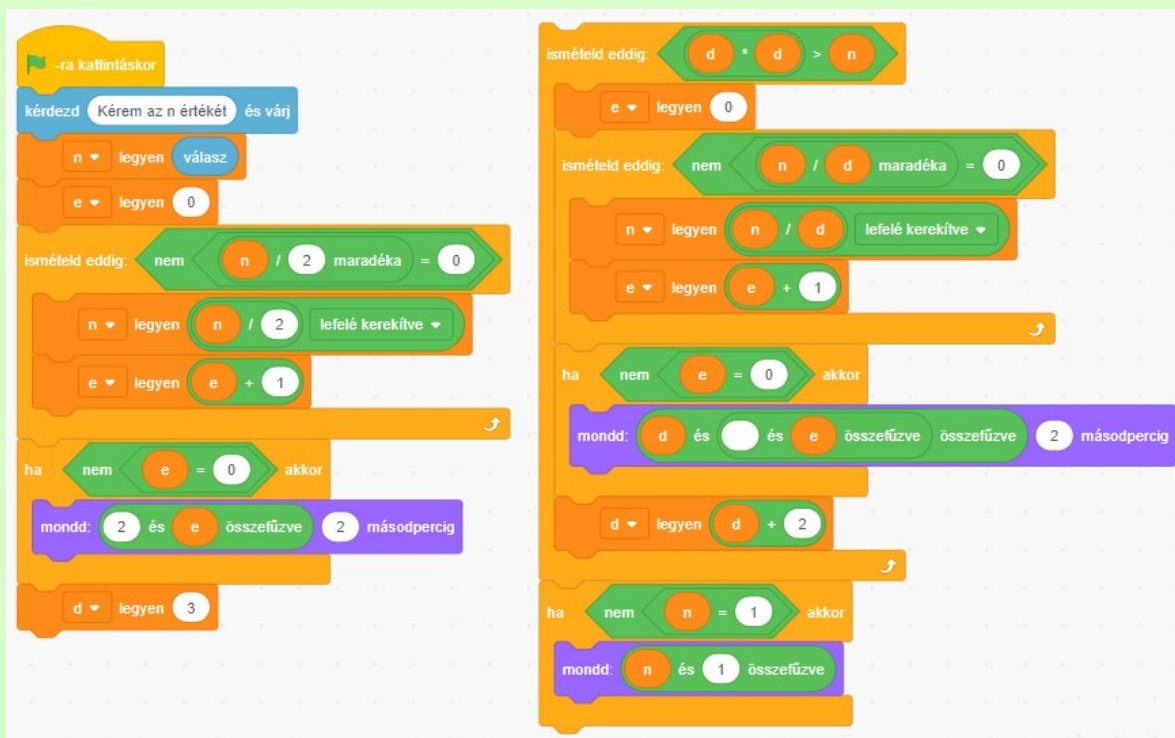


1. Írj C++ programot, amely beolvassa az  $n$  ( $9 < n < 10^6$ ) természetes számot, melynek páros számú számjegye van, és amely ellenőrzi, hogy a szám szimmetrikus-e. Egy számot szimmetrikusnak nevezünk, ha egy természetes szám kétszeri, közvetlenül egymás után való leírásával állítható elő. Például  $n=127127$  egy szimmetrikus szám.
2. Írd át az alábbi C++ programot vele egyenértékűre, **for** helyett **do... while**-t használva:

```

#include <iostream>
using namespace std;
int main()
{
    int n, i, s=0;
    cin >> n;
    for(i=1; i<=n; i++)
        s=s+i;
    cout << s;
}
  
```




3. Írd át a következő algoritmust egyenértékű C++ programra, de használj **do... while...** ismétlődő utasítást minden **ismételd eddig...** ciklus helyett:





## 1. FEJEZET

### 1.1. Lecke

1. **Edit** → **Time/Date**, majd elvégezzük az óhajtott beállításokat.
2. Nem. A Notepad egy szövegszerkesztő.
3. **Ctrl+S**.
4. A menü legördíthető **ALT+** aláhúzott betűvel; például **ALT+e** (**ALT+E**) legördíti az **Edit** menüt.
5. **File**.
6. **Home, View**.
7. Képek beillesztése Wordpad-ben: **Home** → **Insert** csoport →  gomb.
8. **Home** → **Font** csoport →  gomb.
9. Kijelöljük a szöveget; **Home** → **Paragraph** csoport →  gomb. Vagy **Ctrl+E** billentyűzetről.
10. **Home**

### 1.2. Lecke

1. Bármelyik három a következők közül: **Home, Insert, Page Layout, References, Mailings, Review, View**.
2. A **Review** fülön vagy a státuszszoron: Word Count.
3. **Home**.                      4. **ALT+Tab**.
5. A gyorsmenü beállítása (Quick Access Toolbar)
6. Minimalizáljuk a szalagot (**Ribbon**).
7. Igen.                      8. Nem.

### 1.3. Lecke

1. **Ctrl+N, Design** → **Page Color**
2. **Design** → **Page Borders**
3. Írunk egy bekezdést. Ismételjük: **Ctrl+C** másolás, **Ctrl+V** beillesztés, amíg betelik három oldal.
4. **Layout** → **Hyphenation**.
5. **Design** → **Page Color**.
6. Használhatunk szövegdobozt vagy alakzatokat.
7. Igen.
8. Különböző objektumok (képek, táblázatok,...) pontos elhelyezése az oldalon.
9. A kötésre (ragasztásra, kilyukasztásra) szánt extra lapszél beállítása érdekében használjuk.







#### 1.4. Lecke

1. Ctrl+N, Ctrl+S, Ctrl+O.      2. Ctrl+C, Ctrl+V x 5. Alt+F4.

3. Ctrl+F, Ctrl+E.      4.       5. Ctrl+R.

6.  Arial,  18, ,  .      7.  Borders and Shading...


8. Ctrl+E, Ctrl+R, Enter x 2.


9.  Box,  Color: ,  Width: 3 pt ,  ,  
Ctrl+S, Alt+F4, kilépés.


#### 1.5. Lecke


1. PrintScreen, Ctrl+N, Ctrl+S, Ctrl+V, Alt+P, Alt+SZ, Alt+A.


2. Instrumente imagine (Picture Tools) -  Bordură imagine ▾ .


3. Instrumente imagine (Picture Tools) -  Efecte imagine ▾ .

4. Instrumente imagine (Picture Tools) -  Efecte imagine ▾ .

5. Instrumente imagine (Picture Tools) -  Stiluri imagini, Ctrl+S.


6. Instrumente imagine (Picture Tools) -  Efecte artistice ▾, Ctrl+S.

7. Instrumente imagine (Picture Tools) -  .


8. Instrumente imagine (Picture Tools) -  Corecturi ▾ .

9. Ctrl+S, Alt+F4, Kilépés (Exit).


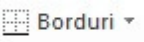
#### 1.6. Lecke

1. Ctrl+N, Ctrl+S,  .

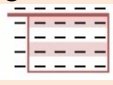
2. Az adatok kitöltése a típus figyelembevételével (karakterek, egész számok, racionális számok).

3. Aspect (Elrendezés, Layout)  Distribuire coloane,  Distribuire rânduri .



4. Aspect lap  ,  Color: ,  Box,  , Ctrl+S.

5. Aspect lap  Sortare gomb.      6. Proiectare lap (Design, Tervezés):  Borduri ▾ .

7. Aspect lap  Margini celulă gomb.      8. Aspect lap  Inserare gomb, Proiectare lap,  gomb.

9. Proiectare lap,  gomb. Ctrl+S, kilépés.

## 2. FEJEZET

1.  - Start recording;  - Pause;  - Stop
2. a. Save movie; b. beszúrsz/hozzáadsz
3. b
4. d
5. b
6. a
7. b
8.  - VLC Media Player;  - Windows Media Player;  - BS Player

## 4. FEJEZET

### 4.1. Lecke

1.  $1 \leftrightarrow f, 2 \leftrightarrow b, 3 \leftrightarrow c, 4 \leftrightarrow e, 5 \leftrightarrow d$
2. a. 2 a b 5, \* + / b. igaz c. hamis
3. a. hamis b. igaz c. hamis
4. a. igaz b. igaz c. igaz d. hamis
5. Szöveg: az összeg, szóköz, az s változó értéke megnövelve 1-gyel
6. 14
7. Az x és y változókba tárolt értékek felcserélése.  $m \leftarrow x; x \leftarrow y; y \leftarrow m$
8. Olvas a, b (természetes számok); Ír  $b-2*a$
9. Olvas x (természetes szám); Ír  $[x/100]*10 + x\%10$
10. Például: Beolvassunk három természetes számot: a,b,c. Írj algoritmust, amely kiírja, hogy ez a három szám képezheti-e egy háromszög oldalhosszait.
11. Például: Beolvassunk három természetes számot: a,b,c. Írj algoritmust, amely kiírja, hogy ez a három szám képezheti-e egy háromszög oldalhosszait. Ha igen, írja ki azt is, hogy milyen a háromszög.
12.  $s \leftarrow$  az a számjegyeinek összege.
13. i.  $cm \leftarrow$  az a legnagyobb számjegye. ii. a.  $cm \leftarrow 9$ ; Figyelj az egyenlőtlenségre! b. Minden számjegyet összehasonlítunk c-vel és számláljuk cm-ben. c. A legnagyobb számjegy minden módosulásakor 1-re állítjuk a számláló értékét, a legnagyobb számjegy minden új értékénél növeljük a számláló értékét.
14.  $a \leftarrow (a,b)$ .
15. i.  $k \leftarrow$  az a változó 0 számjegyeinek száma; ii. a. Csak megszámláljuk a számjegyeket, értékek ellenőrzése nélkül; b. Ellenőrizzük a számjegy paritását (vagyis, hogy páros-e vagy páratlan).
16. i.  $s \leftarrow 5-6+7-8+9-10$ ;  $s \leftarrow a-(a+1)+(a+2)-(a+3)+\dots$ ; ii. Bontsuk fel az előbbi kifejezés zárójeleit és egyszerűsítsünk. Mit veszünk észre? Találj olyan képletet, amelyben csak a és b szerepel, (b-a) paritásától függően.
17. i.  $k \leftarrow$  a páratlan osztóinak száma; ii. a.  $s \leftarrow s+i$ ; b. vajon szükséges-e az ismételd ... ameddig? Figyelj a lépés-re! c. Egy prímszámnak pontosan 2 osztója van. d. Számláljuk kp-be a 2-vel való osztások számát. Majd ... találd ki, kp és k függvényében!

### 4.2. Lecke



1. c. futtatás; d. tesztelés; a forráskód szerkesztése; e hibakeresés; b fordítás
2. a. projekt; b. a **debug** opció

#### 4.3. Lecke

1.  $7/11 \rightarrow 0$ ;  $12\%5 \rightarrow 2$ ;  $7+2 \rightarrow 9$ ;  $5-7 \rightarrow -2$ ;  $17\%10 \rightarrow 7$ ;  $2*3 \rightarrow 6$
2. a. 4; b. 9; c. 86
3. a.  $845/100*10+845\%10$ ;  
b.  $(578\%10)*100+(578/10\%10)*10+578/100$

#### 4.4.3. Lecke

1. d.
2. c.
3. a-2, b-1, c-3

#### 4.4.4. Lecke

```
1. #include <iostream>
using namespace std;
int main()
{
    int a,b,n,c,t,z;
    cin>>a;
    cin>>b;
    t=a*b;
    z=a+b;
    n=t*2+3*z;
    c=n%10+n/10;
    cout<<n<<" "<<c;
}
```

```
2. #include <iostream>
using namespace std;
int main()
{
    int n,a,b,c,S;
    cout<<"Kerem az n erteket: ";
    cin>>n;
    a=n/100;
    b=n/10%10;
    c=n%10;
    S=a+b+c;
    cout<<S;
}
```

```
3. #include <iostream>
using namespace std;
int main()
{
    int n,S;
    cout<<"Kerem az n erteket: ";
    cin>>n;
    S=n*(n+1)/2;
    cout<<"Az osszeg: "<<S;
}
```

```
4. #include <iostream>
using namespace std;
int main()
{
    float x,d;
    cout<<"Ird be a vonat sebesseget km/h-ban: ";
    cin>>x;
    x=x/60;
    d=60/x;
    cout<<"Idotartam percben: "<<d;
}
```

5. 

```
#include <iostream>
using namespace std;
int main()
{
int x,ora,perc;
cout<<"Ird be a film idotartamat percekben: ";
cin>>x;
ora=x/60;
perc=x%60;
cout<<ora<<" ora es "<<perc<<" perc";
}
```

6. 

```
#include <iostream>
using namespace std;
int main()
{
int a,b,c;
float ma;
cin>>a>>b>>c;
ma=(a+b+c)/3.0;
cout<<ma;
}
```

7. 

```
#include <iostream>
using namespace std;
int main()
{
int x,y,z,t;
cin>>x>>y>>z;
t=4*x+2*y+4*z;
cout<<t;
}
```

#### 4.5. Lecke

1. c;
- 2.

```
#include <iostream>
using namespace std;
int main()
{
int a,b,c;
cin>>a>>b>>c;
if((a+b>c) && (a+c>b) && (b+c>a))
{
if((a==b) & (a==c))
cout<<"egyenlo szaru";
else
if((a==b) || (a==c) || (b==c))
cout<<"egyenlo oldalu";
else
cout<<"altalanos";
cout<<" haromszog";
}
else
cout<<"nem haromszog";
}
```

3. a) 3 8, b) 3 7

4.

```
#include <iostream>
using namespace std;
int main()
{
    int a,b,c;
    cin>>a>>b>>c;
    if ((a==b+c) || (b==a+c) || (c==b+a))
        cout<<"lehetseges";
    else
        cout<<"nem lehetseges";
}
```

5.

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    cout<<"Kerem az n erteket:";
    cin>>n;
    if (n%2==0)
        cout<<"A szam paros";
    else
        cout<<"A szam paratlan";
}
```

6.

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    cout<<"Kerem az n erteket:";
    cin>>n;
    if (n>0)
        cout<<"A szam pozitiv";
    if (n==0)
        cout<<"A szam egyenlo nullaval";
    if (n<0)
        cout<<"A szam negativ";
}
```

#### 4.6.1. Lecke

1. a;

2.

```
#include <iostream>
using namespace std;
int main()
{
    int a,k;
    cin>>a;
    k=1;
    while (a>9)
    {
        k=k+1;
        a=a/10;
    }
    cout<<k;
}
```

3.

```
#include <iostream>
using namespace std;
int main()
{
    int a,k=0;
    cin>>a;
    while(a!=0)
    {
        if (a%2==0)
            k=k+1;
        a=a/10;
    }
    if (k==0)
        cout<<"A szam nem tartalmaz paros szamjegyeket";
    else
        cout<<k;
}
```

4.

```
#include <iostream>
using namespace std;
int main()
{
    int n,i;
    cin>>n;
    i=1;
    while (i<=n)
    {
        if(n%i==0)
            cout<<i<<" ";
        i=i+1;
    }
}
```

5.

```
#include <iostream>
using namespace std;
int main()
{
    int n,i;
    cin>>n;
    i=1;
    while (i<n)
        i=i*2;
    if(i>n)
        cout<<i/2;
    else
        cout<<i;
}
```

6.

```
#include <iostream>
using namespace std;
int main()
{
    int n,c,tukor=0;
    cin>>n;
    c=n;
    while(c!=0)
    {
        tukor=tukor*10+c%10;
        c=c/10;
    }
    if(tukor==n)
        cout<<"Tukorszam";
    else
        cout<<"Nem tukorszam";
}
```

7.

```
#include <iostream>
using namespace std;
int main()
{
    int a,b;
    cin>>a>>b;
    while (a!=b)
    {
        if(a>b)
            a=a-b;
        else
            b=b-a;
    }
    cout<<a;
}
```

#### 4.6.2. Lecke

1. d;

2.

```
#include <iostream>
using namespace std;
int main()
{
    int a,b,s=0,nr=1,i;
    cin>>a>>b;
    for(i=a;i<=b;i++)
    {
        s=s+nr*i;
        nr=(-1)*nr;
    }
    cout<<s;
}
```

3.

```
#include <iostream>
using namespace std;
int main()
{
    int n,s=1,i;
    cin>>n;
    for(i=2;i<=n/2;i++)
        if(n%i==0)
            s=s*i;
    if(s==n)
        cout<<"tokeletes szam";
    else
        cout<<"nem tokeletes szam";
}
```



4.

```
#include <iostream>
using namespace std;
int main()
{
    int n,d,i,j;
    cin>>n;
    for(i=n-1;i>1;i--)
    {
        d=0;
        for(j=2;j<=i/2;j++)
            if(i%j==0)
                d++;
        if(d==0)
            cout<<i<<" ";
    }
}
```

5.

```
#include <iostream>
using namespace std;
int main()
{
    int n,ok,i,j;
    cin>>n;
    for(i=n;i>=1;i--)
    {
        ok=0;
        for(j=1;j<=i;j++)
            if(j*j==i)
                ok=1;
        if(ok==1)
            cout<<i<<" ";
    }
}
```

### 4.6.3. Lecke

1.

```
#include <iostream>
using namespace std;
int main()
{
    int a,k;
    cin>>a;
    k=0;
    do
    {
        if(a%10==0)
            k=k+1;
        a=a/10;
    }while(a!=0);
    cout<<k;
}
```

2.

```
#include <iostream>
using namespace std;
int main()
{
    int n,nr=0,c,tukor=0,n1=0,n2=0,p=1,masolat;
    cin>>n;
    masolat=n;
    do
    {
        tukor=tukor*10+n%10;
        nr++;
        n=n/10;
    } while(n!=0);
    c=nr/2;
    do
    {
        n1=n1*10+tukor%10;
        p=p*10;
        tukor=tukor/10;
        nr--;
    }while (nr!=c);
    if(masolat/p==n1)
        cout<<"A szam szimmetrikus";
    else
        cout<<"A szam nem szimmetrikus";
}
```

3.

```
#include <iostream>
using namespace std;
int main()
{
    int n,i,s=0;
    cin>>n;
    i=1;
    if(i<=n)
        do
        {
            s=s+i;
            i++;
        }while(i<=n);
    cout<<s;
}
```

4.

```
#include <iostream>
using namespace std;
int main()
{
    int n, e, d;
    cin >> n;
    e = 0;
    do
    {
        n = n / 2;
        e = e + 1;
    } while (n % 2 == 0);
    if (e != 0)
        cout << 2 << " " << e << endl;
    d = 3;
```

```
if (d * d <= n)
    do
    {
        e = 0;
        if (n % d == 0)
            do
            {
                n = n / d;
                e = e + 1;
            } while (n % d == 0);
        if (e != 0)
            cout << d << " " << e << endl;
        d = d + 2;
    } while (d * d <= n);
if (n != 1)
    cout << n << " " << 1;
}
```

Webhelyek, ahol gyakorolhatsz:

- <https://scratch.mit.edu/> – programozás, gyakorlatok megoldása grafikus blokkokkal
- <https://www.pbinfo.ro/> – programozási feladatok C++ nyelvben
- <https://www.w3schools.in/cplusplus-tutorial/> – C++ segédlet



Informatică și TIC - clasa a VII-a (limba maghiară)