

# JavaScript évfolyam ZH

**Határidő** 2020. nov 6 18:30 -ig      **Pont** 30      **Beküldés...** egy fájlfeltöltés      **Fájltípusok** zip  
**Elérhető** 2020. nov 6, 16:00 - 2020. nov 6, 18:38 körülbelül 3 óra

Ez a feladat zárolva lett ekkor: 2020. nov 6, 18:38 .

## Tudnivalók

- A zárthelyi megoldására **120 perc** áll rendelkezésre. **További 30 percert** adunk az alább olvasható [README.md](#) fájl kitöltésére, a feladatok elolvasására, az anyagok letöltésére, összecsomagolására és feltöltésére.
- A feladatokat a Canvas rendszeren keresztül kell beadni. **A rendszer pontban 18:30-kor lezár, ezután nincs lehetőség beadásra.**
- A feladatok megoldásához **bármilyen segédanyag használható** (dokumentáció, előadás, órai anyag, cheat sheet). A zh időtartamában igénybe vett **emberi segítség tilos** (szinkron, aszinkron, chat, fórum, stb)! Erről nyilatkoztok az alább olvasható [README.md](#) fájlban is, ahol tudomásul veszitek ennek következményeit.
- A feladatok nem épülnek egymásra, **tetszőleges sorrendben** megoldhatók.
- A feladatok megoldásához először [toltsétek le az általunk készített keretprogramot](#). Ebben minden feladat külön könyvtárban helyezkedik el. minden könyvtárban előkészítettük a HTML, CSS, JavaScript állományokat. Ezekben dolgozz! Általában csak a [.js](#) fájlhoz kell hozzájárni, de ha kell, akkor a HTML is módosítható, sőt több [.js](#) fájira is szétoszthatod a megoldásodat, de ez egyáltalán nem elvárás.
- A letöltött keretprogramban lévő [README.md](#) fájlban töltse ki a nevetek és a Neptun azonosítókat (a <> jeleket nem kell beleírni)! **A megfelelően kitöltött [README.md](#) fájl nélkül a megoldást nem fogadjuk el!**

```
<Hallgató neve>
<Neptun kódja>
Webprogramozás - számonkérés
Ezt a megoldást a fent írt hallgató küldte be és készítette a Webprogramozás kurzus számonkéréséhez.
Kijelentem, hogy ez a megoldás a saját munkám. Nem másoltam vagy használtam harmadik féltől
származó megoldásokat. Nem továbbítottam megoldást hallgatótársaimnak, és nem is tettem közzé.
Az Eötvös Loránd Tudományegyetem Hallgatói Követelményrendszeré
(ELTE szervezeti és működési szabályzata, II. Kötet, 74/C. §) kimondja, hogy mindaddig,
amíg egy hallgató egy másik hallgató munkáját - vagy legalábbis annak jelentős részét -
saját munkájaként mutatja be, az fegyelmi vétségnek számít.
A fegyelmi vétség legsúlyosabb következménye a hallgató elbocsátása az egyetemről.
```

- minden feladat könyvtárában találsz egy [TASKS.md](#) fájlt. Ezekben az egyes [\[ \]](#) közötti szóközt cseréld le [x](#)-re azokra a részfeladatokra, amiket sikerült (akár részben) megoldanod! Ez segít nekünk abban, hogy miket kell néznünk az értékeléshez.

## 1. feladat: Haiku szerkesztő (haiku)

Készíts egy haiku szerkesztőt. A haiku egy olyan japán versforma, amely 3 sorból áll, és a soronkénti szótagszám sorrendben 5-7-5. Szerencsére a magyar nyelvben a szavakban lévő szótagok száma minden megegyezik a magánhangzók számával, így viszonylag könnyű ellenőrizni a fenti feltételeket. A haiku szerkesztő egy egyszerű többsoros szöveges beviteli mező ([textarea](#)), amelybe írhatjuk a verset, sorok

végén ENTER-t nyomva. Gépelés közben azonban különböző statisztikákat ír ki a begépelt szövegről, ami ha a fenti formai feltételeket teljesíti, akkor a szerkesztő zöldre vált, és megjelenik egy gomb, amire kattintva a szerkesztőben lévő szöveget egy, az oldalon lévő listába tudjuk felvenni. Feladatok:

- a. Az `index.html`-ben elő van készítve minden olyan HTML elem, amire szükség lehet a megoldáshoz. Mindegyik elemben vannak alapértelmezett értékek, ezeket majd az egyes feladatokkal felül kell írni, de legalább látszik, milyen formában várjuk el őket.
- b. (1 pont) A szöveges beviteli mezőbe gépelve írd ki a mező tartalmát a konzolra! Használ az `input` eseményt!
- c. (1 pont) A szerkesztőben gépelve a `number-of-characters` azonosítójú elembe írd ki a szerkesztőben lévő karakterek számát!
- d. (1 pont) A szerkesztőben gépelve a `number-of-rows` azonosítójú elembe írd ki a szerkesztőben lévő sorok számát (amiket `\n` jel választ el egymástól)!
- e. (1 pont) A szerkesztőben gépelve írd ki a konzolra az első sorban lévő magánhangzók számát! Segítségképpen a magyar nyelv magánhangzói: `aáééííóóööúúúú`.
- f. (1,5 pont) A `vowels-per-row` azonosítójú listában sorold fel, hogy soronként hány magánhangzó szerepel!
- g. (1,5 pont) Ha 3 sor van, és soronként 5-7-5 szótagszám, akkor add a szöveges beviteli mezőt tartalmazó `p` elemnek a `haiku` stílusosztályt. Ekkor zöldre vált a háttér, és megjelenik egy gomb. Ha a feltétel nem teljesül, akkor ne legyen rajta ez a stílusosztály!
- h. (1 pont) Ha jól alkalmaztad a stílusosztályt, akkor haiku sorában megjelent egy gomb. Erre kattintva a szerkesztő tartalmát add hozzá a `haikus` azonosítójú elemhez `<pre>` elemek között (ld. az oldalon lévő példákat)!

**Haiku editor**

Write your poem here:

Number of characters: 0

Number of rows: 1

Number of vowels per row:

1.0

---

**Haikus**

```
hűs hajnal ölel
már halkulnak a tücsök
légy türelemmel

Fölhívtál reggel...
azóta a mosolyod
ben nem mosolyog...

nézlek én Téged
ahogy az ágon fekszel
szerelemes vagyok
```

## 2. feladat: Madármegfigyelés (bird-watch)

Az ország néhány helységében madármegfigyelést végeztünk. Mindegyikben megadtuk, hogy milyen fajú madár ból hányat láttunk. A számolást egy táblázatban kell elvégezni, ahol az  $i, j$ . cella azt tartalmazza, hogy az  $i$ . sorban lévő helységben a  $j$ . madárfajból hányat láttunk. Ha még egy egyedet látunk, akkor a cellába kattintva nő a cellában tárolt darabszám. minden táblázatváltozáskor további feladatokat kell újraszámolni! Figyelem! Nem kell sehol megadni a helység vagy a madárfaj nevét! Ezeket a sor, illetve oszlopindexekkel azonosítjuk!

- a. Az `index.html` állományban elő van készítve egy űrlap, amelyen keresztül megadható a helyek ( $n$ ) és madárfajok ( $m$ ) száma. A "Generate table" gombra kattintva a háttérben létrejön egy  $n \times m$ -es mátrix. Az ehhez tartozó kód az `index.js` állományban van. A további feladatokkal bővítsd az `index.js` tartományt!
- b. (1,5 pont) A "Generate table" gombra kattintva a `table-container` azonosítójú elemben jelenítsd meg JavaScript segítségével a `matrix` változóban tárolt mátrixot HTML táblázatként (`table`, `tr`, `td` elemeket használva)! Pl.  $n=2$  és  $m=2$  esetén ilyesmit kell megjeleníteni.

```
<table>
  <tr>
    <td>0</td>
    <td>0</td>
  </tr>
  <tr>
    <td>0</td>
    <td>0</td>
  </tr>
</table>
```

- c. (1 pont) A táblázat egy cellájára kattintva írd ki a konzolra az adott cella `sor, oszlop` koordinátáját! Használj delegálást, ha tudsz, ld. d. pont!
- d. (1,5 pont) A c. feladatot úgy oldd meg, hogy csak egy elemhez van eseménykezelő kötve (delegálás)!
- e. (1 pont) Az alábbi feladatokat minden cellakattintásra ki kell számolni és meg kell jeleníteni. Ezekhez érdemes tömbfüggvényeket használni, úgy minden feladat 1 sor! Ha nem tudtad volna a b. és c. feladatot megoldani, akkor egy előre beégetett mátrix esetében számold ki az eredményeket és írd a megfelelő elemekbe!
- f. (1 pont) A `task-1` azonosítójú elembe írd ki ("Yes" vagy "No" szöveggel), hogy láttak-e madarat a megfigyeltek közül az első helységben (a mátrix első sorában van-e 0-tól eltérő érték)!
- g. (1 pont) A `task-2` azonosítójú elembe írd ki, hogy hány helységben láttak valamelyik madárfajból 10-nél többet (hány sorban van olyan cella, aminek az értéke nagyobb, mint 10)!
- h. (1 pont) A `task-3` azonosítójú elembe írd ki egy olyan helység sorszámát, ahol a madár se jár (egyik madárfajból se észlelte egyet sem)! Ha nem létezik ilyen helység, akkor ugyanide írd ki, hogy "No"!

## Bird watch

Places:  Species:

### Bird counts



### Questions

Is there any bird in the first place?

(Yes)

In how many places have they seen more than 10 birds from any of the bird species?

(1)

Give one place where no birds have been seen! (row number indexed from 1, "No" if there is no such place)

(2)

## 3. feladat: Hangvizualizáció (canvas-audio)

Hanglejátszást szokták úgy megjeleníteni, hogy a különböző frekvenciartományokban lévő hangerősséget valahogyan ábrázolják, pl. összekötve az egyes értékeket egy törtvonallal. Ebben a feladatban természetesen nem lesz igazi audio, csak szimulálni fogunk egy ilyen jellegű adatot, és készítünk hozzá egy megjelenítést 2d rasztergrafika segítségével.

- a. Az `index.html` állományban elő van készítve egy 210x210 px-es `canvas` elem, kis stíussal megtűzelve, illetve van még két gomb az egyes feladatokhoz! A feladatokat az üres `index.js` állományba készítsd el!
- b. (1 pont) Készíts egy `random(a, b)` függvényt, ami  $[a, b]$  zárt intervallumban állít elő egész számokat! Írj ki egy véletlen számot -10 és 10 között a konzolra!
- c. (1,5 pont) Tölts fel egy 20 elemű tömböt véletlen számokkal -5 és 5 között (határokat beleértve)! A feltöltött tömböt írd ki a konzolra!
- d. (1,5 pont) Rajzold ki a tömb elemeit a vászonra a következőképpen. Húzz egy törtvonalat a vászon bal szélénél közepétől a jobb szélénél közepéig úgy, hogy a bal középső után menjen egy szakasz 10px-szel jobbra és a tömb 1. elemének megfelelően a vászon függőleges közepétől fel vagy le. A következő szakasz innen 10px-re jobbra és a tömb 2. elemének megfelelően a vászon függőleges közepétől fel vagy le, és így tovább. Pl. ha a tömbben ilyen adatok vannak: `[-4, 1, -5, 5, ..., -3]`, akkor a pontok koordinátái:

### Tömb elem X koord Y koord Y koord számolása

	0	105	fix végpont
-4	10	101	105-4
1	20	106	105+1
-5	30	100	105-5
5	40	110	105+5
...	...	...	...
-3	200	102	105-3

**Tömb elem X koord Y koord Y koord számolása**

210      105      fix végpont

- e. (1 pont) A vonal legyen 3px vastag és szürke.
- f. (1,5 pont) A "Change" gombra kattintva minden tömbelemhez adj hozzá egy véletlen számot -1 és 1 között (határokat beleértve), és egy vászontörlés után rajzold is ki az új tömbnek megfelelő állapotot!
- g. (1,5 pont) A "Start animation" ikonra kattintva ezt tessz automatikussá és folyamatossá. Valamilyen időzítő segítségével ([requestAnimationFrame](#) (<https://developer.mozilla.org/en-US/docs/Web/API/window/requestAnimationFrame>) vagy [setInterval](#) (<https://developer.mozilla.org/en-US/docs/Web/API/WindowOrWorkerGlobalScope/setInterval>)) minden változtasd meg a tömbelemeket és törlés után rajzold is ki a vászonra a megváltozott tömböt!

**4. feladat: Haladásnapló (progress-manager)**

Egy nagyobb munkafolyamat független teendőit (pl. könyvírás fejezeteit) kisebb lépésekre bontjuk. A kisebb lépéseket sorrendben kell elvégezni. Ha egy kis lépéssel végeztünk, akkor rákattintva elvégzettnek jelölhetjük. Viszont nem minden azonnal rögzítjük a haladást, így meg kellene oldani, hogy egy kis lépésre kattintva az adott teendőn belüli, de sorrendben előtte lévő, késznek még nem jelölt lépések is késznek legyenek jelölve.

- a. Az [index.html](#) állományban elő van készítve egy teendő lista kisebb lépésekre bontva. A feladatokat az [index.js](#) állományban olld meg!
- b. (1 pont) Egy lépésre kattintva add hozzá a [done](#) stílusosztályt! Ha lehet, már itt használj delegálást, ld. h. pont!
- c. (1 pont) Ha egy teendőre kattintunk, akkor azon ne alkalmazódjon a [done](#) stílusosztály!
- d. (1,5 pont) Ha egy lépésre kattintottunk, akkor az előtte lévő el nem végzett lépések is legyenek ellátva [done](#) stílusosztállyal! Segítség: használható például a [previousElementSibling](#) (<https://developer.mozilla.org/en-US/docs/Web/API/NonDocumentTypeChildNode/previousElementSibling>) metódus a testvéreken való lépkedésre.
- e. (1 pont) Ha egy végzettnak jelölt listaelemre kattintunk, akkor vegyük le a [done](#) stílusosztályt, de csak akkor, ha utána még egyik lépés sincs elvégezve a teendőn belül! Segítség: használható például a [nextElementSibling](#) (<https://developer.mozilla.org/en->

[US/docs/Web/API/NonDocumentTypeChildNode/nextElementSibling](#) metódus a testvéreken való lépkedésre.

- f. (1 pont) Ha egy teendőn belül az összes lépés elvégzett, akkor a teendő is kapja meg a **done** stílusosztályt.
- g. (1 pont) Ha van vagy lesz el nem végzett lépés, akkor a teendő nem lehet kész!
- h. (1,5 pont) A fenti feladatokat úgy old meg, hogy csak egyetlen elemen van regisztrálva az eseménykezelés (delegálás)! Segítség: használható a **parent > child** CSS szelektort a teendők és a lépések megkülönböztetésére.

## Progress manager

- 1. Todo 1
  - Step 1
  - Step 2
  - Step 3
  - Step 4
  - Step 5
- 2. Todo 2
  - Step 1
  - Step 2
  - Step 3
  - Step 4
  - Step 5
  - Step 6
  - Step 7
- 3. Todo 3
  - Step 1
  - Step 2
  - Step 3
  - Step 4
  - Step 5
  - Step 6
- 4. Todo 4
  - Step 1
  - Step 2
  - Step 3



Kritérium	Értékelések		pontok
1/b. A szöveges beviteli mezőbe gépelve írd ki a mező tartalmát a konzolra! Használd az input eseményt!	<b>1.0 pont</b> Teljes megoldás	<b>0.0 pont</b> Nem értékelhető	1.0 pont
1/c. A szerkesztőben gépelve a number-of-characters azonosítójú elembe írd ki a szerkesztőben lévő karakterek számát!	<b>1.0 pont</b> Teljes megoldás	<b>0.0 pont</b> Nem értékelhető	1.0 pont
1/d. A szerkesztőben gépelve a number-of-rows azonosítójú elembe írd ki a szerkesztőben lévő sorok számát (amiket \n jel választ el egymástól)!	<b>1.0 pont</b> Teljes megoldás	<b>0.0 pont</b> Nem értékelhető	1.0 pont
1/e. A szerkesztőben gépelve írd ki a konzolra az első sorban lévő magánhangzók számát! Segítségképpen a magyar nyelv magánhangzói: aáééíóóóőúúű.	<b>1.0 pont</b> Teljes megoldás	<b>0.0 pont</b> Nem értékelhető	1.0 pont
1/f. A vowels-per-row azonosítójú listában sorold fel, hogy soronként hány magánhangzó szerepel!	<b>1.5 pont</b> Teljes megoldás	<b>0.0 pont</b> Nem értékelhető	1.5 pont
1/g. Ha 3 sor van, és soronként 5-7-5 szótagszám, akkor add a szöveges beviteli mezőt tartalmazó p elemnek a haiku stílusosztályt. Ekkor zöldre vált a háttér, és megjelenik egy gomb. Ha a feltétel nem teljesül, akkor ne legyen rajta ez a stílusosztály!	<b>1.5 pont</b> Teljes megoldás	<b>0.0 pont</b> Nem értékelhető	1.5 pont
1/h. Ha jól alkalmaztad a stílusosztályt, akkor haiku sorában megjelent egy gomb. Erre kattintva a szerkesztő tartalmát add hozzá a haikus azonosítójú elemhez <pre> elemek között (Id. az oldalon lévő példákat)!	<b>1.0 pont</b> Teljes megoldás	<b>0.0 pont</b> Nem értékelhető	1.0 pont
2/b. A "Generate table" gombra kattintva a table-container azonosítójú elemben jelenítsd meg JavaScript segítségével a matrix változóban tárolt mátrixot HTML táblázatként (table, tr, td elemeket használva)!	<b>1.5 pont</b> Teljes megoldás	<b>0.0 pont</b> Nem értékelhető	1.5 pont
2/c. A táblázat egy cellájára kattintva írd ki a konzolra az adott cella sor, oszlop koordinátáját! Használj delegálást, ha tudsz, Id. d. pont!	<b>1.0 pont</b> Teljes megoldás	<b>0.0 pont</b> Nem értékelhető	1.0 pont
2/d. A c. feladatot úgy oldd meg, hogy csak egy elemhez van eseménykezelő kötve (delegálás)!	<b>1.5 pont</b> Teljes megoldás	<b>0.0 pont</b> Nem értékelhető	1.5 pont

Kritérium	Értékelések		pontok
2/e. Az alábbi feladatokat minden cellakattintásra ki kell számolni és meg kell jeleníteni. Ezekhez érdemes tömbfüggényeket használni, úgy minden feladat 1 sor! Ha nem tudtad volna a b. és c. feladatot megoldani, akkor egy előre beégetett mátrix esetében számold ki az eredményeket és írd a megfelelő elemekbe!	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
2/f. A task-1 azonosítójú elembe írd ki ("Yes" vagy "No" szöveggel), hogy láttak-e madarat a megfigyeltek közül az első helységben (a mátrix első sorában van-e 0-tól eltérő érték)!	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
2/g. A task-2 azonosítójú elembe írd ki, hogy hány helységben láttak valamelyik madárfajból 10-nél többet (hány sorban van olyan cella, aminek az értéke nagyobb, mint 10)!	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
2/h. A task-3 azonosítójú elembe írd ki egy olyan helység sorszámát, ahol a madár se jár (egyként madárfajból se észlelte egyet sem)! Ha nem létezik ilyen helység, akkor ugyanide írd ki, hogy "No"!	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
3/b. Készíts egy random(a, b) függvényt, ami [a, b] zárt intervallumban állít elő egész számokat! Írd ki egy véletlen számot -10 és 10 között a konzolra!	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
3/c. Tölts fel egy 20 elemű tömböt véletlen számokkal -5 és 5 között (határokat beleértve)! A feltöltött tömböt írd ki a konzolra!	1.5 pont Teljes megoldás	0.0 pont Nem értékelhető	1.5 pont
3/d. Rajzold ki a tömb elemeit a vászonra a következőképpen. Húzz egy törívonalat a vászon bal szélénél közepétől a jobb szélénél közepéig úgy, hogy a bal középső után menjen egy szakasz 10px-szel jobbra és a tömb 1. elemének megfelelően a vászon függőleges közepétől fel vagy le. A következő szakasz innen 10px-re jobbra és a tömb 2. elemének megfelelően a vászon függőleges közepétől fel vagy le, és így tovább.	1.5 pont Teljes megoldás	0.0 pont Nem értékelhető	1.5 pont
3/e. A vonal legyen 3px vastag és szürke.	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
3/f. A "Change" gombra kattintva minden tömbelemhez adj hozzá egy véletlen számot -1 és 1 között (határokat beleértve), és egy vászontörlés után rajzold is ki az új tömbnek megfelelő állapotot!	1.5 pont Teljes megoldás	0.0 pont Nem értékelhető	1.5 pont

Kritérium	Értékelések		pontok
	1.5 pont Teljes megoldás	0.0 pont Nem értékelhető	
3/g. A "Start animation" ikonra kattintva ezt tudd automatikussá és folyamatossá. Valamilyen időzítő segítségével (requestAnimationFrame (Linkek egy külső oldalra) vagy setInterval (Linkek egy külső oldalra)) minden változtasd meg a tömbelemeket és törlés után rajzold is ki a vászonra a megváltozott tömböt!	1.5 pont Teljes megoldás	0.0 pont Nem értékelhető	1.5 pont
4/b. Egy lépésre kattintva add hozzá a done stílusosztályt! Ha lehet, már itt használj delegálást, ld. h. pont!	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
4/c. Ha egy teendőre kattintunk, akkor azon ne alkalmazódjon a done stílusosztály!	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
4/d. Ha egy lépésre kattintottunk, akkor az előtte lévő el nem végzett lépések is legyenek ellátva done stílusosztállyal!	1.5 pont Teljes megoldás	0.0 pont Nem értékelhető	1.5 pont
4/e. Ha egy végzettnek jelölt listaelemre kattintunk, akkor vegyük le a done stílusosztályt, de csak akkor, ha utána még egyik lépés sincs elvégezve a teendőn belül!	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
4/f. Ha egy teendőn belül az összes lépés elvégzett, akkor a teendő is kapja meg a done stílusosztályt.	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
4/g. Ha van vagy lesz el nem végzett lépés, akkor a teendő nem lehet kész!	1.0 pont Teljes megoldás	0.0 pont Nem értékelhető	1.0 pont
4/h. A fenti feladatokat úgy oldd meg, hogy csak egyetlen elemen van regisztrálva az eseménykezelés (delegálás)!	1.5 pont Teljes megoldás	0.0 pont Nem értékelhető	1.5 pont

Összpontszám: 32.0 pontból