# HALMAZOK

Halmazelméleti alapfogalmak, hatványhalmaz,
halmazműveletek, halmazműveletek azonosságai.

## 1. Alapfogalmak

A *halmaz* és az *eleme* fogalmakat alapfogalmaknak tekintjük, nem definiáljuk őket. Jelölés: $x \in H$, azaz $x$ eleme a $H$ halmaznak. Itt $x$ egy tetszőleges valami, mivel a $H$ elemei is tetszőleges dolgok lehetnek. Egy halmaz elemeit megadhatjuk felsorolással, képlettel, körülírással; a lényeg, hogy egyértelműen kiderüljön, hogy mik tartoznak a halmazba, és mik nem. Egy halmaz *véges*, ha véges sok eleme van. Ezt a véges számot a halmaz *elemszám*ának nevezzük. Egy $H$ halmaz elemszámát $|H|$-val jelöljük.

**1. Definíció.** Az **üres halmaz** olyan halmaz, melynek nincs eleme. Jele: $\emptyset$. Másképpen fogalmazva: minden $x$-re teljesül az, hogy $x \notin \emptyset$.

2. *Megjegyzés.* Csak egyetlen üres halmaz van, viszont sok különböző módon felírható. Például

$$\emptyset = \{\text{10-nél nagyobb páros prímszámok}\} = \{\text{4 fejű piros kutyák}\}.$$

**3. Definíció.** Két **halmaz egyenlő**, ha elemeik megegyeznek. Jelölés: $A = B$.

4. *Megjegyzés.* Az előző definíció értelmében egy halmazban minden elemet egyszeres multiplicitással számolunk. Például $\{0,1,2\} = \{0,0,0,1,1,2,2\}$, mert a két oldal elemei ugyanazok. Ugyanígy értelmetlen a halmazban az elemek sorrendjét megkülönböztetni.

**5. Példa.** [1] $\{2,4,6,8\} = \{8,4,2,6\} = \{x \in \mathbb{N} : x < 10 \text{ és } \exists y \in \mathbb{N}, \text{ hogy } x = 2y\} =$ „egyjegyű páros számok halmaza”

**6. Példa.** $\emptyset \neq \{\emptyset\}$
A bal oldali halmaz az üres halmaz, neki nincs eleme, elemszáma nulla. A jobb oldali egy olyan halmaz, mely 1 darab elemet tartalmaz, mégpedig az üres halmazt. Ennek az elemszáma 1. Ez a kettő olyan, mint egy üres könyv, illetve az üres könyvet tartalmazó polc. A könyv üres, de a polc nem. A két halmaz természetesen nem egyenlő.

[1] $\forall x$ jelentése „bármely $x$”, „minden $x$”; $\exists x$ jelentése „létezik $x$”, „van olyan $x$”

## 2. Részhalmaz, hatványhalmaz

**7. Definíció.** Az $A$ halmaz a $B$ halmaznak **részhalmaz**a, ha minden $A$-beli elem egyben $B$-nek is eleme. Jelölés: $A \subseteq B$.

**8. Példa.** $\{1,5,8\} \subseteq \{1,4,5,6,8\} \subseteq \mathbb{N} \subseteq \mathbb{R}$.

9. *Megjegyzés.* Minden $H$ halmaznak van két triviális részhalmaza:

- $H \subseteq H$, azaz minden halmaz részhalmaza saját magának, illetve
- $\emptyset \subseteq H$, azaz az üres halmaz minden halmaznak részhalmaza.

Tulajdonképpen ez a kettő az üreshalmaz esetén egybeesik.

**10. Definíció.** Az $A$ halmaz a $B$ halmaznak **valódi részhalmaz**a, ha részhalmaza, de nem egyenlő vele. Jelölése: $A \subset B$.

**11. Példa.** A 8. Példában szereplő részhalmazjelek tulajdonképpen valódi részhalmazt jelentenek.

A részhalmaz fogalmára érvényesek olyan nyilvánvaló tulajdonságok, mint valós számok körében a „kisebb vagy egyenlő” relációra.

**12. Tétel.** *Legyen $A, B$ és $C$ tetszőleges halmaz. Ekkor*

- *ha $A \subseteq B$ és $B \subseteq C$, akkor $A \subseteq C$;*
- *ha $A \subseteq B$ és $B \subseteq A$, akkor $A = B$.*

13. *Megjegyzés.* Az előző tételben szereplő tulajdonságoknak neve is van, ami a későbbiekben még sokszor elő fog fordulni. Az első tulajdonság azt fejezi ki, hogy a részhalmaz reláció *tranzitív*, a második pedig azt, hogy *antiszimmetrikus*. Az pedig, hogy minden halmaz részhalmaza saját magának azt mutatja, hogy a részhalmaz reláció *reflexív*.

**14. Definíció.** Egy $H$ halmaz **hatványhalmaz**ának nevezzük azt a halmazt, mely a $H$ halmaz összes részhalmazát tartalmazza elemként. Tehát ez egy olyan halmaz, melynek elemei halmazok. Jelölése: $\mathcal{P}(H)$.

**15. Példa.** $\mathcal{P}(\{1,2\}) = \{\emptyset, \{1\}, \{2\}, \{1,2\}\}$.

Általában egy rögzített, jól definiált halmaz elemeivel foglalkozunk, ugyanis nem túl sok értelme van a $H = \{0, 1, 2, 3, \text{Shakespeare összes művei}, p, q, r\}$ halmaznak. Ez is egy korrekten definiált halmaz, de gyakorlati haszna nem túl sok van. Ezért meg szoktunk állapítani egy alaphalmazt, és csak ezen alaphalmaz elemeit vizsgáljuk, az ezen kívüli elemekkel nem foglalkozunk. Például a prímszámok halmazának vizsgálatakor az alaphalmazt tekinthetjük például az egész számok halmazának, mert úgy sem akarjuk azt vizsgálni, hogy egy ceruza eleme-e a prímszámok halmazának. Mivel a ceruza nincs az alaphalmazban, így nem is merül fel ilyen kérdés.

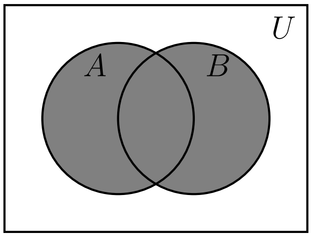
Ha már azonos típusú elemekből álló halmazokat vizsgálunk, akkor bevezethetünk a halmazaink között műveleteket.

## 3. Halmazműveletek

**16. Definíció.** Legyen $A$ és $B$ két tetszőleges halmaz, $U$ legyen a rögzített alaphalmaz, $A, B \subseteq U$. (A formális definíciók mellett a műveleteket Venn-diagramokon is szemléltetjük.)

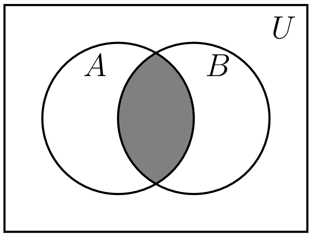
(a) Az $A$ és $B$ halmazok **unió**jának nevezzük azt a halmazt, melynek minden eleme benne van valamelyik halmazban.
Jelölés: $A \cup B$.

$$A \cup B = \{x : \ x \in A \text{ VAGY } x \in B\}$$



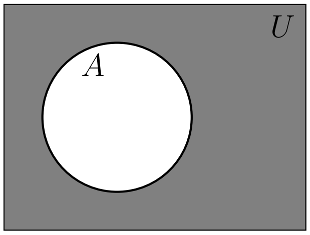

(b) Az $A$ és $B$ halmazok **metszet**ének nevezzük azt a halmazt, melynek minden eleme benne van mindkét halmazban.
Jelölés: $A \cap B$.

$$A \cap B = \big\{x : \ x \in A \text{ ÉS } x \in B\big\}$$



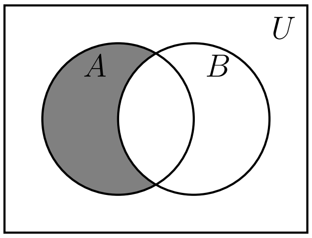

(c) Az $A$ halmaz **komplementer**ének nevezzük azt a halmazt, melynek minden eleme benne van $U$-ban (az alaphalmazban), de nincs benne $A$-ben.
Jelölés: $\overline{A}$.

$$\overline{A} = \big\{x : \ x \in U \text{ ÉS } x \notin A\big\}$$



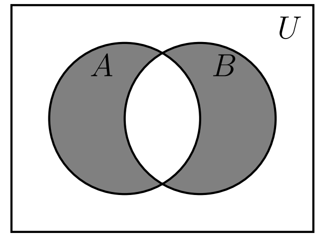

(d) Az $A$ és $B$ halmazok **különbség**ének nevezzük azt a halmazt, melynek minden eleme benne van $A$-ban, de nincs benne $B$-ben.
Jelölés: $A \setminus B$.

$$A \setminus B = \big\{x : \ x \in A \text{ ÉS } x \notin B\big\} = A \cap \overline{B}$$



(e) Az $A$ és $B$ halmazok **szimmetrikus differenciá**jának nevezzük azt a halmazt, melynek minden eleme az $A$ és a $B$ halmazok közül pontosan az egyikben van benne. Jelölés: $A \triangle B$.

$$\begin{aligned} A \triangle B &= (A \setminus B) \cup (B \setminus A) \\ &= (A \cup B) \setminus (A \cap B) \end{aligned}$$




## 4. Halmazműveleti azonosságok

Ebben a részben a halmazműveletek néhány fontosabb tulajdonságát vizsgáljuk meg. Tételként fogunk rájuk hivatkozni, de az állítások legnagyobb része az előbbi definíciók alapján könnyen és gyorsan igazolható.

**17. Tétel.** *Tetszőleges $A, B, C$ halmazokra*

| | | |
|---|---|---|
| $A \cap A = A,$ | $A \cup A = A,$ | *(idempotencia)* |
| $A \cap B = B \cap A,$ | $A \cup B = B \cup A,$ | *(kommutativitás)* |
| $(A \cap B) \cap C = A \cap (B \cap C)\,,$ | $(A \cup B) \cup C = A \cup (B \cup C)\,,$ | *(asszociativitás)* |
| $(A \cup B) \cap A = A,$ | $(A \cap B) \cup A = A,$ | *(abszorptivitás)* |
| $(A \cup B) \cap C =$ | $(A \cap B) \cup C =$ | *(disztributivitás)* |
| $(A \cap C) \cup (B \cap C)\,,$ | $(A \cup C) \cap (B \cup C)\,.$ | |

**18. Tétel.** *Tetszőleges $A, B\,(\subseteq U)$ halmazokra*

$$\overline{A \cap B} = \overline{A} \cup \overline{B}, \qquad \overline{A \cup B} = \overline{A} \cap \overline{B}, \qquad \textit{(de Morgan azonosságok)}$$

$$\overline{\overline{A}} = A,$$

$$A \cap \overline{A} = \emptyset, \qquad A \cup \overline{A} = U,$$

$$A \cap U = A, \qquad A \cup U = U,$$

$$A \cap \emptyset = \emptyset, \qquad A \cup \emptyset = A.$$

A következő tétel már szerepelt a halmazműveletek definíciójánál, azonban fontosságuk miatt tételként is leírjuk újra. A halmazműveletek definícióinál az igazi definíciók a szöveges definíciók, azokból lehet levezetni a következő egyenlőségeket a nyelvtani kötőszavakat megfelelően variálva.

**19. Tétel.** *Tetszőleges $A, B\,(\subseteq U)$ halmazokra*

$$A \setminus B = A \cap \overline{B},$$

*és*

$$A \triangle B = (A \setminus B) \cup (B \setminus A) = (A \cup B) \setminus (A \cap B)\,.$$

Az előző két tétel segíthet abban, hogy a különböző halmazműveleteket átírjuk más halmazműveleti jelek segítségével.

20. *Megjegyzés.* A fenti halmazműveletek mindegyike kifejezhető unió, metszet, és komplementer műveletek segítségével. (Sőt, a metszet és az unió közül elég az egyik a de Morgan szabály miatt.)

**21. Példa.**

$$\begin{aligned}
A \setminus (B \triangle \overline{C}) &= A \cap \overline{(B \triangle \overline{C})} &(1)\\
&= A \cap \overline{((B \cup \overline{C}) \setminus (B \cap \overline{C}))} &(2)\\
&= A \cap \overline{((B \cup \overline{C}) \cap \overline{(B \cap \overline{C})})} &(3)\\
&= A \cap (\overline{(B \cup \overline{C})} \cup \overline{\overline{(B \cap \overline{C})}}) &(4)\\
&= A \cap \big((\overline{B} \cap C) \cup (B \cap \overline{C})\big) &(5)
\end{aligned}$$

(1): Különbség átírása a 19. Tétel szerint.
(2): Szimmetrikus differencia átírása a 19. Tétel szerint.
(3): Különbség átírása a 19. Tétel szerint.
(4): De Morgan azonosság alkalmazása a 18. Tétel szerint.
(5): De Morgan azonosság alkalmazása a 18. Tétel szerint, illetve alkalmazzuk ezen tétel második állítását is.

# 5. Alkalmazások

- Halmaz, mint absztrakt adattípus. LÁSD: Algoritmusok és adatszerkezetek I. kurzus.
- Java programozási nyelv: például Set interfész; AbstractSet, HashSet osztály.
  *Programozó cégeknél szoktak ilyen kérdéseket feltenni, mint például a következő. Van egy tömb, melynek elemei tetszőlegesen nagy egész számok. Listázza azon számokat, melyek a tömbben páratlan sokszor szerepelnek. Mindenki elgondolkozhat, hogy ő hogyan csinálná. Több megoldás is van, a megoldás ötletességét szokták értékelni.*
- Formális nyelvek és számítástudomány:
  - ábécé: előre rögzített, meghatározott jelek általában véges halmaza, például $\Sigma = \{\text{magyar nyelv által használt betűk és karakterek}\}$;
  - az ábécé elemei a karakterek, a karakterekből álló sorozatok a szavak, például Shakespeare minden műve egy-egy szó;
  - $\Sigma^*$ az összes szavak halmaza;
  - $\Sigma^*$ részhalmazai a nyelvek, például Shakespeare összes műve egy nyelv.
  - LÁSD: Bonyolultságelmélet kurzus.

*A Turing-gép talán a számítógép működésének egyik legjobb matematikai modellje. Gondoljunk bele abba, hogy olyan programot kell írnunk, ami egy tetszőleges szóról eldönti, hogy palindrom-e. Nyilván ezt nagyon könnyen meg tudja mindenki csinálni. Azonban ha ezt valaki C-ben megírja, könnyen, még nem feltétlen olyan könnyű ezt elméletben megcsinálni. Például mit hol kell tárolni az algoritmus során, mekkora memóriahelyre lesz szükség, és mennyi ideig fog futni a program. Ilyen kérdéseket vizsgál a bonyolultságelmélet.*

- Reguláris kifejezések: olyan string, amivel meghatározható stringek egy halmaza.
  *Például az „a*” kifejezés jelöli az „a” betűvel kezdődő szavak halmazát. Alapszintű programozásban gyakran van szükség hasonló kifejezésekre, például Linux alatt az egy mappában lévő összes pdf fájl kinyomtatása megtörténhet így:* `lpr *.pdf`*. Nem kell előtte összefésülni, hogy aztán egy darabban ki lehessen nyomtatni, vagy egyesével nyomtatgatni.*
- Fontos kiterjesztés: fuzzy-halmazok. Alkalmazásai: irányítástechnika, mesterséges intelligencia, elektronika. LÁSD: Mesterséges intelligencia kurzus.
  *Matematikailag egy objektum mindig vagy eleme a halmaznak vagy nem. Azonban ez túlságosan leegyszerűsítheti a kategorizálást. Gondoljunk bele, hogy egy arcfelismerő robotot akarunk programozni, melynek az a feladata, hogy megmondja egy emberről ($x$), hogy szép-e ($x \in H$). Azonban nem feltétlen szeretnénk azt, hogy mindenki vagy szép, vagy nem szép legyen, ennél sokkal jobb minden emberhez egy számot hozzárendelni ($\mu(x)$), hogy mennyire szép. És ezzel már nem azt mérjük, hogy $x$ eleme-e $H$-nak, hanem azt, hogy mennyire. Például, ha $\mu(x) = 0.95$, akkor ez közel van az 1-hez, tehát „nagyon benne van a halmazban”.*
- Mandelbrot-halmaz és egyéb fraktálok.
  *Szegedi fejlesztésű szoftver a Xaos, mely segítségével érdekes és szép geometriai alakzatokkal találkozhatunk. Ezeknek neve fraktálok. Például van olyan alakzat, melynek végtelen a kerülete, de véges a területe. (Kicsit furcsa lehet, de igaz, nem gépeltem félre.)*
- Számelméleti halmazok: $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$.
- Biológia: rendszertani kategorizálás.
  *Lehet, bele se gondoltunk eddig, de akárki akármivel foglalkozik, halmazokban gondolkozik. A boltban nem összevissza vannak az áruk pakolva, hanem értelmes részhalmazokra bontva (élelmiszer, tisztítószer, autóalkatrész,...). A programnyelvek objektumorientáltak, vagy nem. Az iskolai érdemjegy egy 5 elemű halmazból kerül ki. (Érdekes, senki sem akar egy vizsgán 7-est kapni, pedig az is olyan szám, mint a többi. Mindenki tudja, hogy az alaphalmaz {1,2,3,4,5}.)*
- Minden területen, mindenféle kategóriába sorolás halmazelméleti feladat. Ujjlenyomat keresése adatbázisban, telefonszám keresése telefonkönyvben, ... - ez mind olyan probléma, mely arra vezethető vissza, hogy egy adott objektum eleme-e egy halmaznak. Gyakorlatban a halmazokon már értelmezve van valami sorrendiségi reláció, így már

nem pusztán matematikai halmazokról beszélhetünk, ahol a halmaz elemeinek sorrendje nem számít. LÁSD: Algoritmusok és adatszerkezetek I. kurzus - Keresési és rendezési algoritmusok.
*Például el lehet gondolkozni a következő feladaton. Van egy egész számokat tároló rendezett tömb, azaz a tömb elemei nagyság szerint sorba vannak rendezve. Eleme-e ennek a tömbnek egy megadott szám? Menjünk végig és keressük meg? Vagy ennél hatékonyabb eljárás is van? A probléma reális, ugyanaz, mintha egy szót keresnénk a szótárban. Aki csinált ilyet, szerintem az optimálisabb módszert is használta, csak fel se tűnt neki.*