

1. feladat (string & mértékegység): Paramétersor normalizálása SI-ra

Feladat: Adott egy egy soros szöveg, kulcs=érték egység párokkal, vegyes tizedeselvásztóval és egységekkel:

"L=2,50 m; b=120 mm; F=0.45 kN; rho=7.85 g/cm³"

Írj programot, amely *stringműveletekkel* feldolgozza a sort és visszaad egy ~~szótárat~~ **listát** SI egységekben:

{{"L": 2.5, "b": 0.12, "F": 450.0, "rho": 7850.0}

["L", 2.5, "b", 0.12, "F", 450.0, "rho", 7850.0]

(m, m, N, kg/m³).

A program végezze el:

- Szétbontás ; és = mentén; `strip()` a szóközök eltávolítására.
- Tizedes vessző cseréje pontra: `replace(",", ".")`.
- Egységek felismerése és konverzió: mm→m, cm³→m³, kN→N, g/cm³→kg/m³.
- Típuskonverziók float-ra; formázott kiírás f-stringgel.

Tipp (string API): `split`, `strip`, `lower`, `replace`, `endswith`, `in`.

2. feladat (string & szabványjel): Csavarjelölés parser és As becslés

Feladat: Készíts függvényt, amely listában kapott csavarjelöléseket elemez:

["M8x1.25-8.8", "M10-8.8", "M6x1-A2-70"]

A kimenet legyen lista szótárakkal (`d_mm`, `pitch_mm`, `class`, `material`, `As_mm2`).

Szabályok:

- Szóközök eltávolítása; kis- és nagybetűk egységesítése.
- Az "x" és az "×" mindkettő elfogadott elválasztó.
- Ha a menetemelkedés hiányzik ("M10-8.8"), tekintsd *durva menetnek* (pl. M10 → 1.5 mm).
- **As** közelítő képlet: $A_s \approx \frac{\pi}{4} (d - 0.9382 p)^2$.

A program végezze el:

- Tokenizálás "-" és "x" mentén; számok kinyerése `isdigit+replace`.
- Alapértelmezett menetemelkedésekhez szótár (pl. {"6":1.0, "8":1.25, "10":1.5}).
- f-string riport minden elemre: `f"M{d} p={p:2f} As={As:1f} mm2".`

Tipp: `split`, `replace`, `find`, `index`, `isdigit`, `join`.

3. feladat (string & normalizálás): Anyagjegyzék sorainak összesítése

Feladat: Adott több soros szöveg tételsorokkal:

"3x csavar M8; 2x anya M8; 5 x rugós alátét Ø8.4; 01x CSAVAR m8"

Normalizáld a sorokat és készíts összesítést (darabszám per termék).

A program végezze el:

- Kisbetűsítés, felesleges szóközök eltávolítása (" " → " ").
- Az "x" és "x" egységesítése; a darabszám nullák levágása.
- Ø jel ("Ø") és pontok kezelése a kulcsban (pl. "rugós alátét ø8.4").
- Szinonimák egységesítése: "csavar m8" ≡ "csavar M8".
- Eredmény: szótár és rendezett, több soros f-string riport.

Tipp: replace, lower, strip, split, zfill, startswith, endswith.

4. feladat (string & mintafelismerés): Próbatest-azonosítók értelmezése

Feladat: Elemezd a próbatest-azonosítókat és számíts közelítő térfogatot forma szerint.

["S235JR-10x50x5 #A12", "AlMg3 3x50x50 #B07", "C45 8x\020 #T1"]

Szabályok:

- Tokenek: anyag, méretek, opcionális # kód.
- "d x b x h" téglatest; "d x \0D" henger (d = hossz, D = átmérő).
- Alakfelismerés tisztán stringből ("\0" vagy "ø", illetve számok száma).

A program végezze el:

- Normalizálás: kisbetűsítés kivéve anyag kódot; "x" egységesítés.
- Méretek kinyerése listába; tizedespont egységesítése.
- Térfogat számítása (mm³), majd liter/ cm³ opció kiírás.
- f-string riport: anyag, típus, méretek, térfogat.

Tipp: split, replace, isalpha, isdigit, any/all, join.

5. feladat (string & report): Fárasztási mérési log parse és LaTeX táblázat

Feladat: Egy több soros logból olvasd ki a fő mezőket és készíts LaTeX táblázatot stringből.

t=0 s; N=0 cycles; sigma_max=0 MPa

t=30 s; N=1500 cycles; sigma_max=220 MPa

t=60 s; N=3000 cycles; sigma_max=220 MPa

A program végezze el:

- Soronként `split(";")` majd kulcs=érték párok feldolgozása (`strip`, `split("=")`).
- Típuskonverziók (idő `float`, ciklus `int`, feszültség `float`).
- Rendezés idő szerint; min, max és átlag számítása a kinyert listákból.
- LaTeX `tabular` környezetet *stringként* építsd fel `join`-nal és f-stringekkel.

Kimenet: A táblázatot és az összegzést (min/max/átlag) a program `print()`-elje ki nyers LaTeX-ként.