

Ütemező – a program dokumentációja

Program: `utemezo.py` · **Nyelv:** Python 3 · **Verzió:** 0.5 (félkész) **Készítette:** egy kolléga, aki átkerült egy másik projektre. **A munkát ti folytatjátok.**

Ez a dokumentáció olyan, amelyet egy valódi cégnél kapnátok, ha egy félkész programot kell befejezni. Előbb olvassátok el, csak utána nyúljatok a kódhoz!

1. Mire való a program?

Egy projekt ütemezését számolja ki és rajzolja le. A feladatokat egy listában adjuk meg: mi a neve, ki a felelős, hány hétig tart, és melyik másik feladat után kezdődhet. A program:

1. kiszámolja, **melyik héten kezdődik** és **melyik héten ér véget** minden feladat;
2. kirajzolja a **Gantt-diagramot** a képernyőre;
3. megmondja, mi a **szűk keresztmetszet** (amire a legtöbb feladat vár);
4. figyelmeztet, ha valaki **túltérhelt** (egy héten túl sok feladata van);
5. a saját teendőket besorolja a **sürgős-fontos mátrixba**.

A mintaadat az órán látott projekt: **Hálózat egy kis irodának.**

2. Futtatás

Hol?	Hogyan?
IDLE (a Pythonnal együtt települ)	Start → IDLE → File → Open → <code>utemezo.py</code> → F5
VS Code	nyisd meg a fájlt → jobb felül a ▶ (Run) gomb
Parancssor	<code>python utemezo.py</code> (abban a mappában, ahol a fájl van)
Ha nincs Python a gépen	nyisd meg az online-python.com oldalt, másold be a teljes kódot, Run

Ha a Gantt-diagramban furcsa jelek jelennek meg a ■ helyett, írd át a program elején: `BLOKK = "##"`.

3. Mit kellene kiírnia? (a kész program kimenete)

```
=== GANTT-DIAGRAM ===
Feladat      Felelős   1 2 3 4 5 6 7 8
Felmérés     Anna      ■ . . . . . . .
Tervezés     Bence     . ■■■ . . . . .
Árajánlat    Csilla    . . . ■ . . . .
Megvalósítás Dani      . . . ■■■ . . . ♦ mérföldkő
Dokumentáció Anna      . . . . . ■■■ .
Bemutató     Bence     . . . . . . ■ ♦ mérföldkő
A projekt hossza: 8 hét

=== SZŰK KERESZTMETSZET ===
Tervezés (Bence): 2 feladat vár rá

=== TÚLTERHELTSÉG ===
Senki sem túlterhelt.

=== SÜRGŐS-FONTOS MÁTRIX ===
1. MOST CSINÁLD: matekdolgozat holnap
2. TERVEZD BE: projekt: igénylista, edzés
3. ADD ÁT / GYORSAN: egy barát kér egy szívességet
4. HAGYD EL: céltalan görgetés
```

A félkész program ezt még nem tudja. Hogy mi hiányzik, a 7. pontban van.

4. A program felépítése

A fájl négy részből áll, fentről lefelé:

Rész	Mi van benne?
Beállítások	BLOKK, URES (a Gantt-diagram jelei), LIMIT (hány feladata lehet egy embernek egy héten)
Adatok	a feladatok lista (a projekt) és a teendok lista (a mátrixhoz)
Függvények	a program „gépei”: mindegyik egy dolgot csinál (lásd az 5. pontot)
Főprogram	a fájl alja: sorban meghívja a függvényeket, és kiírja az eredményt

Egy feladat adatai

Minden feladat egy **szótár** (dictionary): kulcs-érték párok kapcsos zárójelben.

```
{"id": 4, "nev": "Megvalósítás", "felelos": "Dani", "idotartam": 2, "fugg": [2],
"merfoldko": True}
```

Kulcs	Típus	Jelentése	Példa
id	egész szám	a feladat sorszáma	4
nev	szöveg	a feladat neve	"Megvalósítás"
felelos	szöveg	ki csinálja	"Dani"
idotartam	egész szám	hány hétig tart	2
fugg	lista	melyik feladatok után kezdődhet (sorszámok). Üres lista: [] = rögtön kezdődhet	[2]
merfoldko	True / False	a feladat vége mérföldkő-e	True
kezdes, vege	egész szám	ezeket a program számolja ki , nem kell beírni	4, 5

Egy érték elérése: `f["nev"] → "Megvalósítás"`. **Fontos szabály:** egy feladat csak olyan feladattól függhet, amely **előtte** van a listában.

5. A függvények

Függvény	Mit kap?	Mit ad vissza vagy csinál?	Állapot
<code>keres(feladatok, sorszam)</code>	a listát és egy sorszámot	az adott sorszámú feladatot	kész
<code>kezdesek_kiszamitasa(feladatok)</code>	a listát	minden feladatba beírja a kezdes és a vege értéket	kész
<code>feladatok_kiirasa(feladatok)</code>	a listát	kírja a táblázatot	kész
<code>projekt_hossza(feladatok)</code>	a listát	a projekt hossza hétben	TODO 1
<code>gantt_rajzolas(feladatok)</code>	a listát	kirajzolja a Gantt-diagramot	TODO 2, 3
<code>szuk_keresztmetszet(feladatok)</code>	a listát	azt a feladatot, amelyre a legtöbb másik vár, és hogy hány	TODO 4
<code>matrix(teendok)</code>	a teendők listáját	kírja a négy negyedét	TODO 5
<code>tulterheltség(feladatok)</code>	a listát	figyelmeztet, ha valakinek LIMIT-nél több feladata van egy héten	TODO 6

6. Hogyan számolja ki a kezdést? (`kezdesek_kiszamitasa`)

A szabály: **egy feladat a függőségei közül a legkésőbb véget érő után egy héttel kezdődik**. Ha nincs függősége, az 1. héten kezdődik. A vége: `kezdes + idotartam - 1`.

Példa, lépésenként:

Feladat	Függ ettől	A függőségek vége	Kezdés	Vége
1 Felmérés	–	–	1	$1 + 1 - 1 = 1$
2 Tervezés	1	1	$1 + 1 = 2$	$2 + 2 - 1 = 3$
3 Árajánlat	2	3	4	4
4 Megvalósítás	2	3	4	5
5 Dokumentáció	4	5	6	7
6 Bemutató	3, 5	4 és 7 → a nagyobb: 7	8	8

A 3. és a 4. feladat ugyanazon a héten kezdődik: **párhuzamos** feladatok. A 6.-nak meg kell várnia a lassabbat (a dokumentációt).

7. Ismert hiányosságok (TODO-lista)

A kódban mindegyiknél egy # TODO megjegyzés van.

#	Hol?	Mi a gond most?	Mit kell csinálnia?	Nehézség
1	projekt_hossza	mindig 0-t ad, ezért üres a Gantt	a legnagyobb vege érték	könnyű
2	gantt_rajzolas	egyik héten sincs 	 , ha a hét a kezdés és a vége között van	könnyű
3	gantt_rajzolas	nem látszanak a mérföldkövek	◆ mérföldkő a sor végére, ha merfoldko igaz	közepes
4	szuk_keresztmetszet	mindig az 1-es feladatot mondja	a legtöbbet „várt” feladat	közepes
5	matrix	a 2. és a 3. negyed üres	minden teendő a saját negyedébe	közepes
6	tulterheltseg	nincs megírva	hetenként, emberenként számolás	nehéz

Ha egy TODO-t kijavítottatok: írjátok át a táblázatban az állapotát „kész”-re, és a fájl elején a verziószámot (0.5 → 0.6 ...). Egy valódi projektben is a kóddal együtt frissül a dokumentáció.

8. Python-szótár (ha valami ismeretlen)

Amit látsz	Mit jelent?
<code>def nev(...):</code>	függvény: egy elnevezett, újrahasználató programrész
<code>return x</code>	a függvény visszaadja az eredményt
<code>for f in feladatok:</code>	ciklus: a lista minden elemével megcsinálja, ami alatta (behúzva) van
<code>range(1, 9)</code>	a számok 1-től 8-ig (a 9 már nem!)
<code>if ... : / elif ... : / else:</code>	feltétel: ha ... / különben, ha ... / különben
<code>a <= x <= b</code>	x a és b között van (a határokat is beleértve)
<code>and, or, not</code>	és, vagy, nem
<code>[]</code>	lista; <code>.append(x)</code> = hozzáfűz a végéhez
<code>{}</code>	szótár; <code>d[kulcs]</code> = az érték; <code>d.get(kulcs, 0)</code> = az érték, vagy 0, ha nincs ilyen kulcs
<code>f"{x:<15}"</code>	szöveg, amelybe beírjuk x-et, 15 karakter szélesen, balra igazítva
<code># ...</code>	megjegyzés: a gép nem olvassa, az embernek szól