

Programozás feladatok

Felhasznált irodalom: <https://pythonidomar.wordpress.com/>

1. Változók

- a. Melyik állat nevét nem írja ki a program? (Csak a válasz megadását követően ér lefuttatni a programot:))

```
1 cica = 'kutya'
2 kutya = 'egér'
3 egér = 'cica'
4 print(cica)
5 print('kutya')
6 print(egér)
7 print('cica')
```

- b. Mi lesz kiírva a program kimenetének utolsó sorában?

```
1 cica = 'kutya'
2 print(cica)
3 kutya = 'egér'
4 egér = 'cica'
5 print('kutya')
6 cica = 'tyúk'
7 print(egér)
8 kutya = cica
9 print('kutya')
10 print(kutya)
```

- c. Írd ki az alábbi szöveget egy print utasítással! Használd a \n és \t kódokat!

```
Otthoni állataim:
Gubanc kutya
Lila cica
```

- d. Tárold el egy változóban a kutyád nevét és írd ki formázott szöveget használva az alábbi minta szerint!

```
A kutyám neve Gubanc.
```

- e. Írj programot, ami bekéri a felhasználó születési évét, majd kiírja, hogy hány éves most! + Kérje be a program azt is, hogy melyik évben vagyunk és ezzel számoljon!
- f. Írj programot, amelyben a felhasználótól bekérünk két számot és kiírjuk az összegüket, különbségüket, szorzatukat, hányadosukat és négyzetüket!
- g. Adott óra, perc, másodperc hármassal megadott időt váltsd át másodpercre! (Az adatokat nem kell mindenképp a felhasználótól kérni, beírhatod őket a programba is.) Sikeres megoldás után készítsd el a feladat megfordítottját!
- h. Írj olyan programot, amelyik a felhasználótól megkérdi, hogy hány magyar mérföldre van a sárkány barlangja, és megmondja ezt kilométerben és tengeri mérföldben (mert ugye lehet, hogy hajózni is kell)! Az osztás jele a per (/). Feltételezheted, hogy a sárkány egész számú magyar mérföldre lakik. A tizedesvessző helyett tizedespontot kell használnod. A törtszámokat két tizedesre kerekítve írd ki!
- i. Kihívást jelentő feladat: Milyen szöveget zár be egymással a kis és a nagy mutató adott időpontban? Kérj be egy időpontot, majd írd ki a szöveget fokban és radiánban is (ha nem tudod, mi az, nézz utána a neten)

2. Elágazások

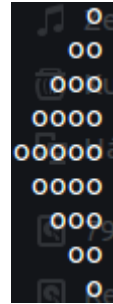
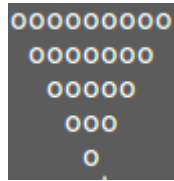
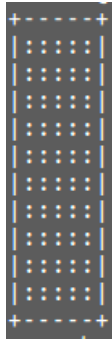
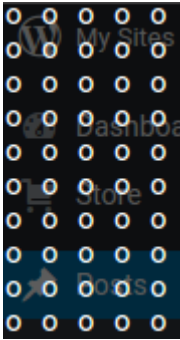
- a. Írj olyan programot, amelyik jelszót kér a felhasználótól! Ha a felhasználó tudja a jelszót, akkor „bemehet”, különben „hozzáférés megtagadva” üzenetet adja.

- b. Írj ki két, változóban tárolt számot és kérd be az összegüket! Ha a felhasználó eltalálja az összegüket, dicsérd meg, ha nem, akkor szidd össze keményen.
- c. Írj programot, ami egy szoba méreteiből kiszámolja a területét!
 - + Ha 30 m²-nél nagyobb a terület, írd ki, hogy „Ez egy xx m²-es nagy szoba.”, ha 10 és 30 m² közötti, akkor „Ez egy xx m²-es közepes szoba.”, egyébként „Ez egy xx m²-es kis szoba.” (Az xx helyére mindenhol a felhasználó által megadott érték jelenjen meg!)
 - + A program kérje be a burkoláshoz használt csempe méreteit, majd számolja ki, hogy leburkolható-e egész csempékkel. Ha igen, add meg, hogy hány csempére lesz szükség!
 - + Menjünk biztosra, számolj rá 10% veszteséget, és azt, hogy egy dobozban 12 csempe van. Add meg, hogy hány doboz csempét kell rendelnünk (csak egész dobozzal rendelhetünk)
- d. Kérdezd meg, hogy a felhasználó hányadik lett a versenyen! Ha benne van az első háromban, írd ki, hogy milyen érmet kapott. Csak egy if-sort használhatsz, de elif-et annyit, amennyit úri kedved diktál.
- e. Írjunk programot, ami egy pontról megállapítja, hogy melyik síknegyedbe esik!
- f. Írjunk programot másodfokú egyenlet megoldására!
- g. Döntsük el két számról, hogy osztható-e egyik a másikkal. A felhasználó tetszőleges sorrendben adja meg az osztandót és az osztót, tehát arra is figyelniünk kell, hogy melyik szám nagyobb.

3. Ciklusok

- a. Írj olyan programot, ami kiírja a számokat 1-től 10-ig, és az 5-nél kisebbeket duplán írja ki!
 - b. Írj programot, ami megkérdi a felhasználótól, hogy hánytól hányig írja ki a számokat és aztán írja is ki őket!
 - + Bővítsd a programodat úgy, hogy kérdezze meg azt is, hogy hányasával írja ki a számokat!
 - c. Írj programot, ami kiírja az első száz pozitív négyzetszámot és köbszámot az alábbi formában (a hatványozás jele a két csillag, azaz kettő a köbön: 2**3)!
- | n | n ² | n ³ |
|-----|----------------|----------------|
| 1 | 1 | 1 |
| 2 | 4 | 8 |
| 3 | 9 | 27 |
| ... | ... | ... |
| 45 | 2 025 | 91 125 |
| ... | ... | ... |
- d. Írjuk ki a Fibonacci számokat 1000-ig! Ha nem tudod, mik a Fibonacci számok, nézz utána!
 - e. Számoljuk ki π pontos értékét a $\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots$ összefüggés alapján!
 - f. Írjunk autószimulátort! A program minden parancs után írja ki, hogy éppen jár-e a motor. Reagáljon az „indít” és a „leáll” parancsokra!
 - g. Írjuk kő-papír-olló játékot!
 - + A játék 3 fordulóból áll, a program írja ki, hogy hányadik fordulónál tartunk, és összesítse a pontszámokat!
 - h. Írjunk programot, ami generál egy véletlen számot és a felhasználónak ki kell találnia! A szám egy adott intervallumban legyen, amit konstansként definiálj a programodban és ezt írd ki segítségként a felhasználónak!
 - + Adj visszajelzést, hogy a tipp kisebb vagy nagyobb a gondolt számnál!
 - + Írd ki, hogy mennyire közel jár a felhasználó (meleg, langyos, hideg)

- + Ha nem találja ki az intervallum hossza darab próbálkozásból, akkor jelezd vissza, hogy nagyon béna!
- + Figyelmeztess előre, ha már csak 5 próbálkozásnál kevesebb van hátra!
- i. Döntsük el egy számról, hogy prímszám-e!
 - + Írd ki 100-ig az összes prímet!
- j. Keressük meg két szám legnagyobb közös osztóját és legkisebb közös többszörösét!
- k. Írd ki ciklusokkal az alábbi mintákat:



- l. Oldd meg az $x^3 - 78x^2 + 1787x - 11070 = 0$ egyenletet a természetes számok halmazán! Most nem a megoldóképletet használjuk, hanem egy ciklussal kell próbálgatnod. Annyit segítünk, hogy a harmadfokú egyenletnek legfeljebb 3 gyöke lehet.
 - + Az $x^3 - 58x^2 + 427x + 11070 = 0$ egyenletnek negatív gyöke is van. Módosítsd úgy a programodat, hogy ezt is képes legyen megtalálni. Itt is számíthatsz rá, hogy három különböző gyököt találsz.
 - + **Ábrázold a függvényt derékszögű koordinátarendszerben!**
- m. Oldd meg a $3x + 4y = 52$ egyenletet, keresd meg az összes megoldást a természetes számok halmazán! Két egymásba ágyazott ciklust érdemes használnod, a külső az egyik változót pörgeti, a belső a másikat.
- n. Bergengócia piacán 102 tallérért almát vettünk. A jonatán 15 tallérba, a golden 12 tallérba kerül darabonként. Melyikből hány darabot vásároltunk? (Ez a feladat szinte ugyanaz, mint az előző, csak egyenletté kell alakítanod a szöveget)
- o. **Számoljuk ki újra π értékét! Definíció szerint ez az állandó a kör kerületének és átmérőjének hányadosa. Most viszont azt használjuk ki, hogy a kör területének képlete $t = \pi \cdot r^2$. A számításhoz véletlen pontokat fogunk „dobálni” egy egység oldalú négyzetbe (random könyvtár). Mindegyikről eldöntjük, hogy a beírt körben vannak-e vagy sem. Megszámoljuk őket, aztán már nagyon egyszerű kiszámolni π értékét.**
 - + **Rajzold is ki, a pontokat! Színezd zöldre a körön belüli, pirosra pedig a körön kívüli pontokat!**
- p. **Szimuláljuk a bolygók mozgását! Írj olyan programot, ami két égitestet mutat és a gravitációs erő alapján folyamatosan módosítja a sebességüket! Készíts programot, ami bemutatja a mozgásukat!**

4. Megszámolás, összegzés

- a. Szimuláljunk kockadobást! Írj programot, ami egy 100 elemű listát feltölt véletlenszerűen 1 és 6 közötti egész számokkal, majd kiírja a lista tartalmát! Számoljuk meg, hogy hány 6-ost dobtunk és írjuk ki!
 - + számoljuk meg, hogy hány páros / páratlan / prímszám dobásunk volt
 - + számoljuk meg, hogy hányszor fordult elő, hogy 1-es után 2-es jött

- + számoljuk meg, hogy hányszor fordult elő, hogy három dobásban mindig nagyobbat dobtunk, mint az előző
- + számoljuk össze minden lehetőségre 1-től 6-ig, hogy hányszor fordul elő
- b. Generáljuk véletlenszámokkal egy érettségi dolgozat pontlistáját (0-120 pont), adjuk meg, hogy hány jeles született (a ponthatár 96 pont, ezt a program elején konstansként add meg).
 - + számoljuk meg, hogy hány esetben hiányzott 3 pontnál kevesebb az ötöshöz
 - + hány maximális pontszámú dolgozat született
 - + számoljuk meg 10%-os sávonként a dolgozatok számát
 - + számoljuk ki a pontszámok átlagát (szórását)
- c. Ismerjük egy bolt mindennapos kiadását és bevételét. Adjuk meg a veszteséges napok számát!
 - + Hány olyan nap volt, amikor a bevétel legalább 10%-kal nagyobb volt a kiadásnál?
 - + Hány olyan nap volt, amikor az előző naphoz képest nőtt a bolt profitja?
 - + Mennyi volt a teljes időtartamra a profit?
 - + Mennyi volt a hétvégi napok nyeresége, ha az adathalmaz hétfővel indul?
- d. Adott egy kör a középpontjával és a sugarával, valamint egy pontsorozat. Számoljuk meg, hogy hány pont esik a körvonalon belülre!
- e. Adjuk meg egy szöveg magánhangzóinak a számát!
 - + Számoljuk meg, hányszor szerepel benne a „gy” betű
- f. Számoljuk meg, hány prímszám esik adott intervallumba (pl. 5 és 200 között)!

5. Minimum és maximum számítás

- a. Olvasd be az atlétikai diákolimpia magasugrás eredményeit. A fájl két év adatait tartalmazza. Egy sor ugyanannak a versenyzőnek a tavalyi és idei eredményét mutatja. Ha a szám helyén „NA” szöveget látsz, az azt jelenti, hogy a versenyző az adott évben nem indult a diákolimpián.
 - + Számold meg, hogy hány versenyző indult mindkét versenyen, hányan indultak csak tavaly és hányan csak idén!
 - + Add meg, hogy hányan ugrottak valamelyik évben 180 cm fölött. Add meg, hogy hányan ugrottak mindkét évben 180 cm fölött!
 - + Add meg, hogy hányan javítottak a tavalyi eredményhez képest!
 - + Mennyi volt a legmagasabb és a legalacsonyabb ugrás az egyes években?
 - + Mennyi volt a legkisebb és legnagyobb változás a két év között?
- b. Egy napon félóránként mértük a hőmérsékletet.
 - + Számold meg, hogy hány alkalommal volt a hőmérséklet 30°C fölött!
 - + Határozd meg a napi átlaghőmérsékletet!
 - + Számold meg, hogy hány alkalommal nőtt és hány alkalommal csökkent a hőmérséklet az előző méréshez képest!
 - + Add meg a napi legkisebb és legnagyobb hőmérsékletet, és a mérésük időpontját!
 - + Add meg, hogy mikor emelkedett legjobban a hőmérséklet két mérés között!
 - + A hőmérő adatai néha hibásan érkeznek. Gyanúsak azok a mérések, amelyek nagyon eltérnek a két szomszédjuk átlagától. Határozd meg a leggyanúsabb mérést! (forrás: mester.inf.elte.hu – Gyanús hőmérsékletek)

Beolvastam 46 rekordot.

A nap folyamán 5 alkalommal mértünk 30°C-nál nagyobb hőmérsékletet.

A napi átlaghőmérséklet 26.88°C

20 alkalommal nőtt és 25 alkalommal csökkent a hőmérséklet az előző méréshez képest.

A legalacsonyabb hőmérsékletet 2:30-kor mértük és 23.21°C volt.

A legmagasabb hőmérsékletet 15:00-kor mértük és 31.55°C volt.
A legnagyobb emelkedés 2:30-3:00 között volt: 24.63°C - 23.21°C = 1.42°C
A leggyanúsabb mérést 15:30-kor mértük: 31.55, 29.74, 30.43

6. Eldöntés

- a. Egy egyetem 5 éves szakjának végzettjeiről nyilvántartják a születés dátumát (év, hó, nap), és a szak kezdésének, valamint a diplomázásának évét. A bemeneti állomány első sorában a hallgatók száma ($1 \leq N \leq 100$), alatta soronként egy-egy végzett hallgató adatai vannak, öt pozitív egész szám formájában egy-egy szóközzel elválasztva. Az első, második és harmadik szám a végzett születési idejének év ($1950 \leq \text{É} \leq 2000$), hó ($1 \leq H \leq 12$) és nap része ($1 \leq \text{Nap} \leq 31$), a negyedik a kezdés éve ($2000 \leq K \leq 2020$), az ötödik pedig a végzés éve ($K \leq V \leq 2020$). (forrás: mester.inf.elte.hu – Legfiatalabban végzett hallgató)
- + Kérj be a felhasználótól egy életkor értéket, majd dönts el, hogy van-e olyan hallgató, aki ennél fiatalabban szerzett diplomát! (eldöntés)
 - + Van-e olyan hallgató, akinek ugyanazon a napon van a születésnapja, mint Neked? (eldöntés)
 - + Határozd meg, hogy mikor született az a végzett hallgató, aki legfiatalabban szerezte meg a diplomáját! (min/max)
 - + Számold meg, hogy hány hallgató végzett 2016-ban! (megszámlálás)
 - + Mekkora az átlagéletkor az egyetemen 2014 tavaszán? (összegzés, megszámlálás)
- b. Egy városban forgalomszámlálást végeztek különböző helyeken (a mérési helyek számát jelöljük M-mel). Minden áthaladó járműről feljegyezték a megfigyelés helyét és időpontját. A megfigyelési adatokat összegyűjtötték és az időpont szerint sorrendbe rendezték. A bemeneti állomány első sorában két egész szám van: a megfigyelési helyek száma ($2 \leq M \leq 100$) és az adatok száma ($2 \leq N \leq 1000$). A következő sorok mindegyike egy megfigyelési adatot tartalmaz: az első szám a megfigyelési hely sorszáma ($1 \leq SZ \leq M$) a második a megfigyelési időpont óra ($0 \leq \text{Ó} \leq 23$), a harmadik pedig a perc része ($0 \leq P \leq 59$).
- + Írj programot, amely megadja, hogy mi a leghosszabb idő-intervallum kezdete és vége, amikor a két időpont között sehol sem haladt át jármű! (min/max)
 - + Oldd meg a feladat úgy is, hogy az egyes mérési pontokat külön vizsgáljuk, tehát pl. az 1-es pont időintervallumába nem számítjuk bele, ha a 2-es ponton haladt át jármű! (min/max)
 - + Számold meg, hogy hány mérési érték tartozik az 50-es mérési ponthoz! (megszámlálás)
 - + Kérj a felhasználótól egy időpontot és add meg, hogy volt-e mérési eredmény abban az időpontban! (eldöntés)
 - + Volt-e olyan időpont, amikor két mérési eredmény is volt? (eldöntés)
 - + Készíts összesítő statisztikát arról, hogy melyik állomáson hány mérés történt! (megszámlálás)
 - + Melyik állomáson történt a legtöbb mérés? (min/max)

7. Keresés, kiválogatás

- a. Egy ingatlanforgalmazó cég tárolja az eladó lakások alapterületét és árát. A bemeneti fájl első sorában a lakások száma van ($1 \leq N \leq 1000$), a következő N sorban pedig egy-egy lakás alapterülete (négyzetméterben, $1 \leq T \leq 500$) és ára van (millió forintban, $1 \leq \text{Á} \leq 1000$).
- + Írd ki a legdrágább lakás sorszámát és árát! Ha több megoldás van, akkor közülük a legkisebb sorszámút kell kiírni. (min/max)

- + Ellenőrizd, hogy van-e 500M forintnál drágább ingatlan! (eldöntés)
- + Keresd meg azt a lakást, aminek legmagasabb a négyzetméter ára (azaz az ár/terület aránya)! (min/max)
- + Számold meg, hogy hány 20M Ft alatti ingatlan van! (megszámlálás)
- + Írd ki az 50 és 60 m² közötti területű ingatlanok területét és árát az 'árak.txt' nevű fájlba! (kiválogatás)
- + Kérd be a felhasználótól, hogy milyen ár-tartományban keres ingatlant és listázd ki neki! (kiválogatás)
- + Kérj be a felhasználótól egy terület és egy ár értéket és keress neki ilyen ingatlant. Ha nincs ilyen, akkor jelezd ezt neki, ha találsz, akkor listázd ki őket! (kiválogatás)

8. Rendezés

- a. Egy szövegfájlban kártyalapok értékei vannak felsorolva úgy, hogy a lapok értékét 1 és 13 közötti számok reprezentálják (azt is mondhatnánk, hogy 1 és 13 közötti számok vannak benne). Írj programot, ami növekvő sorba rendezi ezeket, majd kiírja a képernyőre és egy fájlba!
- b. Egészítsd ki a korábbi lakás árakra vonatkozó feladatot azzal, hogy sorba rendezed az adatokat a lakások mérete szerint csökkenő sorrendbe, majd a 10 legnagyobbat kiírod egy fájlba az alábbi minta szerint:

1. alapterület:	500 m ² ;	ár:	239 MFt
2. alapterület:	499 m ² ;	ár:	178 MFt
3. alapterület:	499 m ² ;	ár:	80 MFt

10. Számok formázása

Szám	Formázás	Kimenet	Leírás
3.1415926	{:.2f}	3.14	2 tizedes helyig
3.1415926	{:+.2f}	+3.14	2 tizedes helyig jelöléssel
-1	{:+.2f}	-1.00	2 tizedes helyig jelöléssel
3.1415926	{:.0f}	3	Nincs tizedes hely (kerekítve)
5	{:0>2d}	05	Nullával kezdődik az elején, balra
1000000	{:,}	1,000,000	A számformátumban vesszővel választjuk el az ezreket
0.25	{:.2%}	25.00%	Százalékos alak
1000000000	{:.2e}	1.00e+09	Exponenciális alak
11	{:>10d}	11	Jobbra igazított
11	{:<10d}	11	Balra igazított
11	{:^10d}	11	Középre igazított

11. Dátum kezelés

Dátum változó használata

```
import datetime

most = datetime.datetime.now()          # mai dátum és jelenlegi időpont
print(most.year)                        # év
print(most.month)                       # hónap
print(most.day)                         # nap
print(most.hour)                       # óra
print(most.minute)                     # perc
print(most.second)                     # másodperc
régen = datetime.datetime(1956, 10, 23)
napok = most - régen                    # a két dátum közötti különbség
print(napok.days)                      # a különbség napokban megadva
print(most.strftime('%Y.%m.%d'))        # formázott kiírás
```

Legfontosabb időformátumok

Formázó sztring	Jelentés	Példa
%a	a hét napjának rövidített alakja	Sun, Mon, ...
%A	a hét napjának teljes alakja	Sunday, Monday, ...
%w	a hét napjának sorszáma	0, 1, ..., 6
%d	a hónapon belüli nap előnullázva	01, 02, ..., 31
%-d	a hónapon belüli nap	1, 2, ..., 30
%b	hónap rövidített alakja	Jan, Feb, ..., Dec
%B	hónap teljes alakja	January, February, ...
%m	hónap sorszáma előnullázva	01, 02, ..., 12
%y	év utolsó két jegye előnullázva	00, 01, ..., 99
%Y	év értéke négy számjegyen	2013, 2019, ...
%H	óra értéke	00, 01, ..., 23
%M	perc értéke	00, 01, ..., 59
%S	másodperc értéke	00, 01, ..., 59
%f	másodperc törtrésze mikroszekundumban előnullázva.	000000 - 999999
%c	a lokalizációs beállítás szerint formázott teljes dátum és idő	Mon Sep 30 07:06:05 2013
%x	a lokalizációs beállítás szerint formázott dátum	09/30/13
%X	a lokalizációs beállítás szerint formázott idő	07:06:05

Magyar lokalizáció beállítása

```
import locale
locale.setlocale(locale.LC_ALL, 'hu_HU')
```


12. Lista műveletek

```
számok = [1, 2, 3, 4, 5]
számok[0]          # eredménye 1, a lista első eleme
számok[1]          # eredménye 2, a lista második eleme
számok[-1]         # eredménye 5, a lista utolsó eleme
számok[-2]         # eredménye 4, a lista utolsó előtti eleme
számok.append(6)   # a 6-ot hozzáfüzi a lista
számok.insert(0, 6) # a 6-ot beszúrja a listába, 0 indexű elemként
számok.remove(6)   # törli a listából a 6-ot
számok.pop()       # törli a lista utolsó elemét
számok.clear()     # törli a lista minden elemét
számok.index(8)    # eredménye a 8-as érték első előfordulásának indexe
számok.sort()      # rendezi a lista elemeit
számok.reverse()   # megfordítja az elemek sorrendjét
számok.count(3)    # megszámlálja, hogy hányszor fordul elő a listában a 3-as érték
len(számok)        # megadja, hogy hány elemű a lista
sum(számok)        # összeadja a lista elemeit
```

13. Fájl műveletek

```
# ha egy sor egy adat rekordot tartalmaz
adatok = []
forrás = open('fájlnév.txt', mode='r', encoding='utf-8')
for sor in forrás:
    adat = sor.strip().split(';')
    adatok.append([adat[0], adat[1]])
...
forrás.close()
```

```
# ha az első sorban a rekordok száma van
adatok = []
forrás = open('fájlnév.txt', mode='r', encoding='utf-8')

forrás.readline()
for sor in forrás:
    adat = sor.strip().split(';')
    adatok.append([adat[0], adat[1]])
...
forrás.close()
```

```
# kiírás fájlba
kimenet = open('fájlnév.txt', mode='w', encoding='utf-8')
kimenet.write(f'Tudok számolni: 11 + 12 = {11 + 12}.')
kimenet.close()
```