

PYTHON

1. forduló

Starschema
Part of **HCLTech**

A kategória támogatója: Starschema

Ismertető a feladatlaphoz

Kérjük, hogy a feladatlap indítása előtt mindenképp olvasd el az alábbi útmutatót:

MINDEN kérdésre van helyes válasz.

Olyan kérdés NINCS, amire az összes válasz helyes, ha mégis az összes választ bejelölöd, arra a feladatra automatikusan 0 pont jár.

Több válaszlehetőség esetén a helytelen válasz megjelölése mínusz pontot ér.

A radio button-os kérdésekre egy helyes válasz van.

Ha lejár a feladatlap ideje, a rendszer AUTOMATIKUSAN beküldi azt az addig megjelölt válaszokkal.

Azokat a feladatlapokat, amelyekhez csatolmány tartozik, javasoljuk NEM mobilon elindítani, erre az érintett feladatlapok előtt külön felhívjuk a figyelmet.

Az adatbekérős feladatokra NEM jár részpontszám, csak a feleletválasztósakra.

Helyezéseket a 4. forduló után mutatunk, százalékos formában: adott kategóriában a TOP 20-40-60%-hoz tartozol.

Ha egyszerre több böngészőből, több ablakban vagy több eszközről megnyitod ugyanazt a feladatlapot, nem tudjuk vállalni az adatmentéssel kapcsolatban esetlegesen felmerülő anomáliákért a felelősséget!

A ChatGPT használata nem tiltott, de az arra való hivatkozással észrevételt NEM fogadunk el!

A feltűnően rövid idő alatt megoldott feladatlapok kizárást vonnak maguk után, bármilyen más gyanús esetben fenntartjuk a jogot a forduló érvénytelenítésére!

Jó versenyzést kívánunk!

Essünk túl a kategória nehezen az első fordulóban!



Köszöntünk a versenyben! Az idei Python-szekcióban a kezdő és a haladó Pythonosok számára is szeretnénk érdekes feladatokat kínálni. Ezek egy része a nyelv ismeretére, részleteire helyezi a hangsúlyt, máskor viszont a Pythont hasznos, sokoldalú eszközként kell alkalmazni, és ötlet is kell a megoldáshoz.

A kitűzött feladatok egy részében programot kell írni. Bármilyen kódszerkesztő vagy IDE használható (pl. Pycharm, VSCode, Anaconda). Szükséged lehet egyes csomagokra (már a kezdetektől: **pandas**, **numpy**, **re**), ezek se árt, ha kéznél vannak.

A feladatlapok elején egy általános, a Python használatát, esetleg különlegességeit firtató kérdés áll. E kérdések legtöbbször feleletválasztós típusúak.

A feladatlapok többi kérdése programozással kapcsolatos. Egy vagy több kérdésre kell választ adni. Figyelj arra, hogy a megfelelő formátumban küldd be az eredményt!

A feladatokhoz háttértörténet is tartozik: Flórián fiatal informatikus, aki nagyon szereti a sört. Szenvedélye nem áll meg a fogyasztásnál, készíteni is szeretne, sőt, mi több, hősünk elhatározta, hogy hamarosan sörfőzdet fog nyitni. Ám, mint minden vállalkozáshoz, ehhez is sokféle előkészítő vizsgálatot kell végezni. Feladatod ebben segíteni Flóriánnak.

Tanács: A szükséges adatfájlokat töltsd le, ha kell, tömörítsd ki, mielőtt elindítod a feladatlapot, így időt nyersz. Az adatfájlok általában excel formátumúak vagy pusztán szövegesek. Egyes adatokat képként adtunk meg. A verseny vége felé kicsit különlegesebb előkészítésekre lesz szükség..., de ne szaladjunk ennyire előre, mindent a maga idejében!

Az első feladatlap két feladatot tartalmaz.

Jó munkát!

Indítás előtti csatolmányok

Melyek a helyes állítások az alábbi kódrészletek **True** / **False** értékére vonatkozóan?

1.

```
id("this_is_a_test") == id("this" + "_" + "is" + "_" + "a" + "_" + "test")
```

2.

```
bool("F" "a1 " "\bs" "\u0065")
```

3.

```
a = "python"; b = "python"; c = "python!"; d = "python!"  
  
a is b and c is d
```

4.

```
True is False == False
```

Válaszok

- ☐ Az 1. kifejezés értéke: **True**.
- ☐ A 2. kódrészlet kiértékelésekor **False** értéket kapunk.
- ☐ A 2. kódrészlet kiértékelésekor **True** értéket kapunk.
- ☐ A 3. kódrészlet kiértékelésének végeredménye implementációfüggő. Egyes esetekben **True**, más esetekben **False**.
- ☐ A 4. kódrészlet kiértékelésének eredménye: **True**.

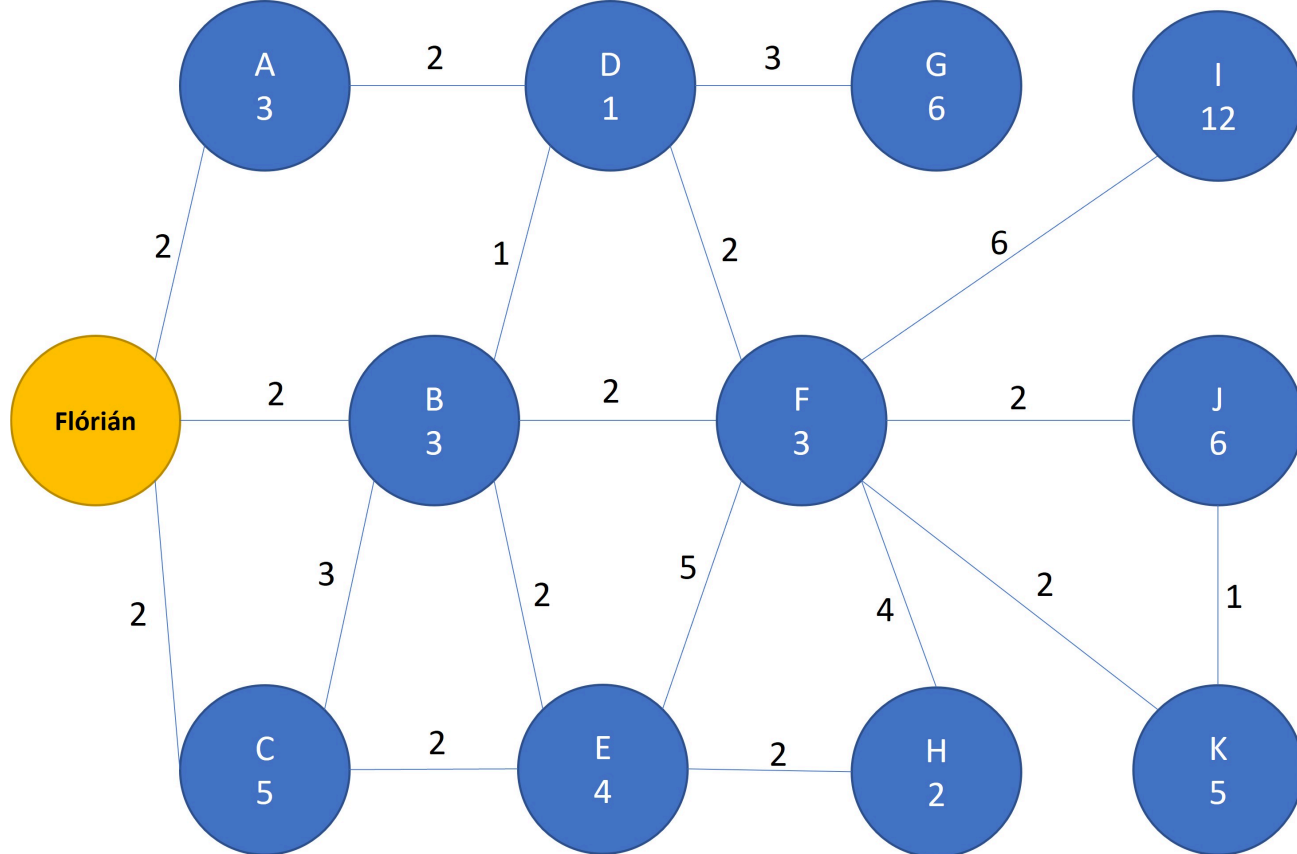
2. feladat 10 pont

Sörtúra

Flóriánnak először is tájékozódnia kell a piacról, ezért sörkóstoló túrát tervez. 24 óra alatt minél több sört szeretne kipróbálni, az alábbi peremfeltételekkel:

Egyfajta sörből egy korsóval szeretne inni, ezt egy szempillantás alatt megteszi;

Rendelkezésére áll egy térkép:



Flórián jelenlegi pozíciójával (FL állomás)

a sörfőzdek közti utakkal és az időbeli távolságokkal (órában);

a sörfőzdekben kapható különböző sörök számával;

az utak oda-vissza járhatók, egy állomás többször is meglátogatható (bár ekkor több sört már nem iszik);

A 24. órában még fel tud hörpinteni egy sört.

Pontosan 24 óra múlva Flórián elalszik a rengeteg sörtől, amit megivott.

Legfeljebb hány sört tud meginni Flórián, és mi az optimális útvonal? Készíts Python programot, amely kiszámítja a végeredményt.

A választ az alábbi formátumban kérjük:

[maximális_sörök_száma:állomás,állomás,állomás,...], azaz szögletes zárójelek között add meg először a kiszámított értéket, majd írd kettőspontot, ezt követően írd le az állomások betűjelét (nagybetűvel) a bejárás sorrendjében, a kiinduló állomással kezdve, az állomásokat egymástól vesszővel elválasztva (szóköz nélkül). A kiinduló állomás jele: FL. Tehát valami ilyesmit várunk:

[2000:FL,A,B,X,Y,Z]

Válaszok

