



ÓBUDAI EGYETEM
NEUMANN JÁNOS INFORMATIKAI KAR

KOCKANAP 2022

12. EGYETEMI 24 ÓRÁS PROGRAMOZÓI VERSENY
(MELLÉKFELADATOK)

<http://users.nik.uni-obuda.hu/szabozs/K24/>

IDŐBEOSZTÁS*

*a szervezők az időpont változás jogát fenntartják

2022.11.04. PÉNTEK

- 15:00 – 16:00 A versenyzők érkezése, regisztráció, előkészületek
- 16:00 – 16:15 Megnyitó, feladatok ismertetése
- 16:15-től A feladat(ok) megoldása (fejlesztési időszak)
- 20:00 – 22:00 Vacsora

2022.11.05. SZOMBAT

- 08:00 – 10:00 Reggeli
- 12:00 – 14:00 Ebéd
- 16:15 – 17:00 **Az elkészült feladatok bemutatása**
- 17:15 Eredményhirdetés, a díjak átadása

SZABÁLYZAT

- A szervezők minden csapat részére biztosítanak 3 db elektromos hálózati végpontot és 1 db számítógépes hálózati végpontot (UTP kábelekről és a hálózat tovább osztásához szükséges eszközökről - router/switch- a csapatnak kell gondoskodnia).
- A csapatok számára a rendezők korlátozott Internet elérést biztosítanak a belső hálózatán keresztül.
- A feladat(ok) megoldásához a csapatok tetszőleges szoftvereket felhasználhatnak, amelyeket a verseny előtt vagy közben telepítettek a számítógépekre (a szervezők ehhez segítséget és telepítő csomagokat nem biztosítanak).
- A verseny időtartama alatt a versenyzők nem hagyhatják el a verseny helyszínét, ez ugyanis a csapat azonnali kizárásával jár (dohányzásra kijelölt helyet biztosítunk a helyszínen).
- A verseny időtartama alatt a versenyzők nem fogadhatnak vendégeket az épületben, ez szintén a csapat azonnali kizárásával jár.
- A verseny előtt, alatt és azt követően a résztvevőknek be kell tartaniuk a szervezők (így a helyszínt biztosító Óbudai Egyetem) által adott, a versennyel kapcsolatos szervezési, részvételi és szakmai instrukciókat. Amennyiben valamely résztvevő ennek a kötelezettségének nem tesz eleget, a versenyből további indoklás nélkül azonnal kizárható, és a versennyel kapcsolatos semminemű juttatásra nem jogosult.
- A kihirdetett végeredmény megtámadásának lehetőségét a szervezők nem biztosítják.
- A verseny során elkészített feladatok megoldását (forráskód, futtatható állomány) a versenyzők kötelesek átadni a szervezőknek. Ellenkező esetben az elkészített feladat nem kerül értékelésre.
- A (külön)díjak kiosztásának minimális feltétele a működő alkalmazás.
- A jelentkezéseket a beérkezés sorrendjében fogadjuk el. A szervezők fenntartják a jogot, hogy a hiányos vagy bármely más szempont alapján nem megfelelő jelentkezéseket a jelentkezés bármely fázisában elutasítsa.
- A versenyen való részvétel feltétele, hogy a szervezők elektronikusan visszajelezzék a csapat jelentkezését.
- A versenyből automatikusan kizárásra kerülnek azok a csapatok, akik a feladat(ok) vagy részfeladat(ok) megoldásának részét vagy egészét közzéteszik, illetve a másik csapatot akadályozzák valamilyen formában a feladat elkészítésében, leadásában.

LETÖLTHETŐ ANYAGOK

Az oldalt kérjük rendszeresen nézzétek, mert minden lényeges információ itt fog először megjelenni!

<http://users.nik.uni-obuda.hu/szabozs/K24/>

MELLÉKFELADATOK

Mellékfeladatok csak akkor kerülnek értékelésre, ha sikeresen tudtok csatlakozni a szerverhez és tudjátok irányítani a játékot, illetve a beadási határidő előtt beküldésre kerül a megoldás.

A mellékfeladatok esetében a bemenete(ke)t vagy a Console-ról vagy egy fájlból kell beolvasni és a megoldás(oka)t vagy a Console-ra vagy egy fájlba kell kiírni (*minden feladatnál külön meghatározásra kerül*). Mivel a feladatok kiértékelése automatikusan történik, így fontos, hogy nem szabad a kelletténél több adatot beolvasni vagy kiírni, ugyanis ezesetben a megoldást hibásnak tekintjük. (A kódolási stílust nem vizsgáljuk, bármilyen formában elkészíthető a megoldás, illetve a dokumentáció sem szükséges.)

A mellékfeladatok beadási határideje: 12:00.

A mellékfeladatok megoldását a következő programozási nyelven kell elkészíteni:

C# (.NET Framework 4.7.2)

Minden mellékfeladat rendelkezik egy „képességgel”, ami elnyerhető a megoldásával. Minden feladat többször is megoldható, de minden esetben csak az utolsóként leadott megoldás kerül figyelembevételre (még abban az esetben is, ha az rosszabb eredményt jelent)! Az egyes feladatok beadását követően további nem nyilvános tesztesetekkel is tesztelésre kerül az alkalmazás, elkerülve azokat az eseteket, mikor a versenyző az algoritmus implementálása helyett csak a bemenetet vizsgálva írja ki a megfelelő kimenetet.

A „képességek” segítségével a FőFeladatban lévő tulajdonságok erősíthetők, amik a megszerzést követően a verseny végéig aktívak maradnak nincsen sem idő, sem egyéb korlát a felhasználására (egyszerre több, akár az összes képesség is aktív lehet egy időben).

A Mellékfeladatok megoldásainak és kiértékeléseinek menete:

- Az 'A' csapat megoldja az egyik mellékfeladatot.
- A következő oldalon feltöltik a megoldás forráskódját (egyetlen .cs fájl, aminek elnevezése: csapatnév_feladatneve.cs):

<http://users.nik.uni-obuda.hu/ff/>

- A szervezők kiértékelik a feladatot és az alábbi oldalon közzéteszik az eredményt:

<http://users.nik.uni-obuda.hu/szabozs/K24/>

A Mellékfeladatok és a képességek:

Minden feladat leadását követően lefuttatásra kerülnek az adott feladathoz tartozó tesztesetek. A futtatás eredményét egy százalékos érték fogja mutatni, ami azt jelzi, hogy a tesztesetek hány százalékát teljesítette a beadott algoritmus. Ha az eredmény 100%-os, vagyis minden teszteseten átmént a beadott megoldás, akkor a csapat megkapja a feladatért járó képességet.



I. INFINITE ENERGY
+% ENERGY

megnöveli az ásások és a lövések számát



II. FORCE FIELD
+% SHIELD

megnöveli a pajzs értékét



III. WORMHOLE
+% SIGHT

megnöveli a belátható környezetet

A feladatok megoldásához a szervezőség jó munkát kíván!



I. Mellékfeladat

+% ENERGY

Jack O'Neill vezetésével a CSK-1 egy eddig ismertlen bolygó felfedezésére indult, küldetésük azonban nem várt fordulatot tartogatott, mikor hazafelé vették az irányt. A csillagkaput működtető vezérlőben üzemzavar keletkezett, így a kapu beindításához szükséges energiát nem tudta előállítani. Daniel Jackson próbálta megfejteni a vezérlő működését, mikor Samantha Carter rájött, hogy az eszköz a rendszerben található csillagok energiáját használja működése során. A kapu beindításához nincs más dolguk, mint megadni a csillagokból kinyerhető legtöbb energiát...

Az eszköz, ami képes kinyerni a csillagok energiáját N darab csillagot lát a rendszerben, melyeket 0-tól $N-1$ -ig tart számon, az azokból kinyerhető energia mennyiségével ($N[i]$ az i . csillagból kinyerhető energia).

Az eszköz azonban nem az érintett csillag energiáját nyeri ki, hanem annak túlhevítésével annak szomszédjai által kibocsátott energiát használja fel. Ehhez az alábbi műveleteket hajtja végre:

1. Kiválaszt egy (i .) csillagot, ami nem a legelső és nem a legutolsó az általa ismert csillagok közül.
2. A kiválasztott (i .) csillagot túlhevíti (...a *Tau'ri-k* számára valami ismeretlen *Asgard* technológiával).
3. A csillag túlhevülésével a szomszédos ($N[i-1] * N[i+1]$) csillagok energiáját begyűjti.
4. Az eszköz újra inicializálja a megmaradt csillagokat 0-tól $N-1$ -ig, elkerülve a csillag újbóli túlhevítését, ezzel a megsemmisítését.

Mivel a vezérlő minden esetben a csillagokból kinyerhető legtöbb energiát biztosította a kapunak, ezért a CSK-1 feladata is azonos, vagyis megadni legtöbb összesített energiát, ami a csillagok túlhevítésével és energiájuk kinyerésével előállítható.

Bemenet (fájl; starsystem.in):

A bemeneti fájl egy csillagrendszerben található csillagkapukat és az energia ellátásukhoz szükséges csillagokat tartalmazza. Minden csillagkapu [. .] résszel kezdődik és egy üres sorral végződik. A zárójelek között a csillagkapu neve található, amit követően egyetlen sorban pontosvesszővel (;) elválasztva találhatók a csillagok energiái.

Kimenet (fájl; starsystem.out):

A kimeneti fájlba a bemeneti fájlban található csillagkapukhoz tartozó csillagokból kinyerhető maximális energia mennyiségét kell kiírni soronként elválasztva az alábbi formátum szerint:

csillagkapu neve: a csillagokból kinyerhető maximális energia

Megkötés(ek):

a csillagkapu neve csak az angol ábécé kis- és nagybetűit, illetve szóközt tartalmaz

$3 \leq$ a rendszerben található csillagkapuk száma ≤ 100

$3 \leq$ egy csillagkapu vezérlője által kezelt csillagok száma ≤ 50

$1 \leq$ egy csillagból kinyerhető energia $\leq 1\,000$

Példa (0. TESZTESZET):

INPUT

```
[EARTH]
1;2;3;4

[Abydos]
100;999;100

[Chulak]
100;3;1;2;100

[Edora]
2;2;2;2;2;2;2;2;2;2

[Kheb]
87;9;5;4;75
```

OUTPUT

```
EARTH: 12
Abydos: 10000
Chulak: 10400
Edora: 32
Kheb: 7575
```

Az **EARTH** bolygón található csillagkapuhoz a vezérlő négy darab csillagból kinyerhető energiát használ fel, melyek értékei: 1;2;3;4. Az első és az utolsó csillag a vezérlő feltételei szerint nem hevíthető túl, ezért két lehetőségünk van:

- vagy elsőnek választjuk a csillagot 2 energiával, aminek a szomszédjaiból $1*3=3$ energiát nyerünk ki, majd választjuk a csillagot 3 energiával, aminek a szomszédjaiból $1*4=4$ energiát nyerünk ki, így összesen $3+4=7$ energiához jutunk, vagy
- vagy elsőnek választjuk a csillagot 3 energiával, aminek a szomszédjaiból $2*4=8$ energiát nyerünk ki, majd választjuk a csillagot 2 energiával, aminek a szomszédjaiból $1*4=4$ energiát nyerünk ki, így összesen $8+4=12$ energiához jutunk.

A két lehetőség közül a második esetben jutunk több energiához az **EARTH** esetében, így a kimenet ezt az értéket fogja tartalmazni az alábbi formában: **EARTH: 12**.

Az **Abydos**-on található csillagkapu vezérlője három darab csillag energiáját tudja felhasználni, melyek energiája: 100;9999;100. Mivel itt csak egyetlen csillag hevíthető túl, így más lehetőségünk nincsen, a fájlban a kapuhoz tartozó kimenet az alábbi lesz: **Abydos: 10000**.

A **Chulak** esetében öt csillag energiájából kell meghatározni a lehető legtöbb összesített energiát, amit úgy kaphatunk meg, hogy elsőnek választjuk a 2 energiával rendelkező csillagot ($1*100=100$), majd választjuk az 1 energiával rendelkező csillagot ($3*100=300$), végül az egyetlen megmaradt, 3 energiával rendelkező csillagot ($100*100=10000$), amiket összegezve a fájlba az alábbi kerül: **Chulak: 10400**.

Az **Edora**-n található csillagkapu vezérlője tíz darab csillag energiáját tudja felhasználni, ami alapján a fájlba az **Edora: 32** kerül.

Végezetül az utolsó bolygó a **Kheb** aminek öt csillagjából nyerhető összesített energia 10400, így a kimeneti fájl, a csillagkapuhoz tartozó sora: **Kheb: 7575**.

Nyilvános tesztesetek:

1. TESZTESET

INPUT

[EARTH]

1;2;3;4

[Abydos]

100;999;100

[Chulak]

100;3;1;2;100

[Edora]

2;2;2;2;2;2;2;2;2;2

[Kheb]

87;9;5;4;75

OUTPUT

Earth: 12

Abydos: 10000

Chulak: 10400

Edora: 32

Kheb: 7575



II. Mellékfeladat

+% SHIELD

A CSK-1 sikeresen lopta el Apophis pajzsát, ami, habár működőképes a szimbionta nélkül is, valódi erejét csak azzal együtt tudná kihasználni. Mivel ennek beültetését egyikük sem vállalja, így megoldást kell találniuk a problémára...

A pajzs működésének alapja, hogy a talajtól hozza létre a gazdatest fejéig az erőteret, ami megóvjá őt a sérülésektől. Az erőter létrehozása során, azonban egyes részek kimaradnak, melyeket a Goa'uld felismer és pótol, enélkül az erőter összeomlana. Mivel szeretnénk nélkülözni a parazitát, így a hiányzó elemek pótlásáról az általunk készített szerkezetnek kell gondoskodnia. A pótlás során a létrejött erőter pontjait kell biztosítani arról, hogy azok ne hulljanak a földre, ezzel mintegy alátámasztva azokat.

Például, ha az alábbi erőter jön létre, ahol az 'O' a gazdatest védendő pontjait mutatja ahol létrejött az erőter, a '-' azokat a pontokat, ahol nincsen erőter, akkor pontosan 8 darab helyen kell pótolni az erőteret (amik '*' -al jelölt helyeken történnek meg).

A hibás erőter

```
-O-O---
--OO-OO
-----O
-O---O-
```

A javított erőter

```
-O-O---
-*OO-OO
-***-*O
-O**-O*
```

Bemenet (Console):

N sor: az erőter által generált sor, minden sor pontosan ugyanannyi (W) karaktert tartalmaz

Kimenet(Console):

1 szám: a szükséges pótlások száma az erőteren annak stabilitása érdekében

Megkötés(ek):

$1 \leq N, W \leq 100$

minden sor pontosan ugyanannyi karaktert tartalmaz

a soron belül csak 'O' és '-' karakter szerepel

Példa (0. TESZTESZET):

INPUT

```
--O---O-
```

OUTPUT

0

Public test cases:

1. TEST CASE

INPUT

```
--O---O-
```

OUTPUT

0

2. TEST CASE

INPUT

```
-O-O---
--OO-OO
-----O
-O---O-
```

OUTPUT

8

3. TEST CASE

INPUT

```
-O-O-O-O-O-O-
O-O-O-O-O-O-O
-O-O-O-O-O-O-
O-O-O-O-O-O-O
```

OUTPUT

16

4. TEST CASE

INPUT

```
OOOOO
-OOO-
--O--
-----
```

OUTPUT

11



III. Mellékfeladat

+% SIGHT

A Tau'ri a Daedalus megépítését követően elkészítette első olyan csillagkaput, ami az űrhajók számára is képes féreglyukat nyitni, tekintettel arra, hogy így még mindig gyorsabban lehet haladni, mint amit a hajtóművekkel elérhetnek. Azonban nem várt előnye lett a csillagkapunak, hogy más fajok is használni kezdték...

Az űrhajók számára készült csillagkapu a munkáját a 0. időpillanattól kezdi, ahova minden **M** percben egy új űrhajó érkezik (vagyis az űrhajók 0, M, 2*M, 3*M, ... időpillanatokban érkeznek).

Az érkezéshez hasonlóan, a 0. időpillanattól **N** percenként nyílik meg a féreglyuk, amin áthalad az összes megérkezett űrhajó, mely még mindig a csillagkapu előtt várakozik. Ha egy űrhajó pontosan ugyanabban az időpillanatban érkezik, amikor a féreglyuk megnyílik, akkor azon azonnal át is halad. Ellenkező esetben meg kell várni a következő időpontot, amikor megnyílik a csillagkapu.

A csillagkaput vezérlő személyzet számára fontos, hogy tájékoztassák az érkező hajókat, hogy mennyi az ideig kell várakozniuk átlagosan a féreglyukba lépés előtt ezért ezt a hajók érkezésének és a féreglyukak megnyílásának idejéből ki is számolják.

Például, ha $M=2$ és $N=4$, vagyis 2 percenként érkezik egy űrhajó és 4 percenként nyílik meg a féreglyuk, akkor az űrhajók a 0, 2, 4, 6, 8, ... percben érkeznek és a 0, 4, 8, ... percben haladhatnak át a csillagkapun. Vagyis az űrhajók várakozási ideje: 0, 2, 0, 2, 0, ... ami alapján az átlagos várakozási idő az áthaladás előtt 1.

Bemenet (Console):

1. sor: az űrhajók érkezésének ideje (**M**)
2. sor: a féreglyuk megnyílásának ideje (**N**)

Kimenet(Console):

- 1 szám: az űrhajók átlagos várakozási ideje (1 tizedesjegyre kerekítve) a kapu előtt, mielőtt beléphetne a féreglyukba

Megjegyzés(ek):

Legyen **V(i)** az **M***i. percben érkezett űrhajó várakozási ideje. Az átlagos várakozási idő a következők szerint határozható meg: $(V(0) + V(1) + \dots + V(P-1)) / P$, ahol **P** egy pozitív végtelen szám.

Megkötés(ek):

$$1 \leq N, S \leq 1\,000\,000\,000$$

Példa (0. TESZTESET):

INPUT
4
2

OUTPUT
1.0

Public test cases:

1. TEST CASE

INPUT
4
2

OUTPUT
1.0

2. TEST CASE

INPUT
3
5

OUTPUT
2.0

3. TEST CASE

INPUT
1
6

OUTPUT
2.5

4. TEST CASE

INPUT
2345
12345

OUTPUT
6170.0