

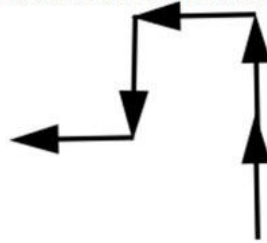
2022.05-digitális kultúra programozás feladat

5. Robot

Sokféle tevékenységet végeznek a környezetünkben az automaták, illetve a robotok.

Egy egyszerű robot a végrehajtandó mozgását egy betűkből álló sorozattal, szóval kapja. Vízszintes síkban szabadon mozog, iránytűje segítségével pontosan be tudja tájolni magát. Az *E*, *D*, *K* vagy *N* betűk hatására egységnyit megy észak, dél, kelet vagy nyugat felé.

Például, ha a robot által kapott szó: *EENDN*, akkor a robot útvonala:



Készítsen programot, amely a parancsszót, azaz a betűk sorozatát egyszerűsíti, vagyis olyan új parancsszót állít elő, amelynek végrehajtásakor a robotot a kezdőpontból a végpontba a lehető legkevesebb utasítással juttatja el! A parancsszavak hossza legfeljebb 200 betű.

| | Parancsszó | Új parancsszó |
|----------|------------|---------------|
| 1. példa | ENEK | EE |
| 2. példa | EENDN | ENN vagy NNE |
| 3. példa | EEKDKDNN | |

A program forráskódját mentse *robot* néven! A program megírásakor a felhasználó által megadott adatok helyességét, érvényességét nem kell ellenőriznie, és feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek. Ha a robot kezdőpontja és mozgásának végpontja azonos, akkor az új parancsszó üres.

A képernyőre írást igénylő részfeladatok esetén az ékezetmentes kiírás is elfogadott.

1. Olvassa be és tárolja el a robot mozgását vezérlő szót, és annak felhasználásával oldja meg a következő feladatokat
2. Írja ki, hogy az egyes betűkből hány darab van a szóban!
3. Írja ki a képernyőre a bekért útvonal egy lehetséges egyszerűsítését, tehát egy olyan új parancsszót, amelyet végrehajtva a robot a lehető legkevesebb mozgással juthat el a kiindulási pontból az eredeti parancsszónak megfelelő végső helyzetbe!

Minta a szöveges kimenet kialakításához:

```
Kérem a robot parancsait: EEEKDKEKDKEKDDNN
E betűk száma: 5
D betűk száma: 4
K betűk száma: 5
N betűk száma: 2
Egy legrövidebb út parancsszava: KKKE
```

A 2022.05-ös Digitális Kultúra érettségi feladat tökéletes arra, hogy megértsd az alapokat: **adatbevitelt, számlálást, logikai gondolkodást** és a **kiíratást**.

Képzeld el a robotot egy négyzethálós papíron. Ha lép egyet fel (Észak), majd egyet le (Dél), akkor visszajutott ugyanoda. A célunk az, hogy a felesleges "oda-vissza" mozgásokat kitöröljük.

Íme a megoldás lépésről lépésre:

1. Lépés: Adatbekérés és tárolás

A programnak először meg kell kérdeznie a felhasználótól a parancsszót. Ezt az `input()` függvénnyel tesszük meg, és egy **változóba** mentjük.

Python

1. feladat: Beolvasás

```
utvonal = input("Kérem a robot parancsait: ")
```

Magyarázat:

Az `utvonal` nevű "dobozba" elmentjük azt a betűsorozatot (szöveget), amit a felhasználó begépel.

Változó (utvonal):

Képzeld el egy felcímkézett dobozt. Amikor a felhasználó beírja a billentyűzeten, hogy "EENDN", a Python ezt a szöveget beteszi az `utvonal` nevű dobozba.

Függvény (input):

Ez egy parancs a Pythonnak: "Állj meg, várj a felhasználóra, és amit beír, add át nekem."

2. Lépés: Betűk megszámlálása

Meg kell néznünk, hányszor szerepel az E, D, K és N betű.

A Pythonban van egy beépített segítség erre: a `.count()` metódus.

Python

2. feladat: Számlálás és kiírás

```
e_db = utvonal.count('E')
```

```
d_db = utvonal.count('D')
```

```
k_db = utvonal.count('K')
```

```
n_db = utvonal.count('N')
```

```
print(f'E betűk száma: {e_db}')
```

```
print(f'D betűk száma: {d_db}')
```

```
print(f'K betűk száma: {k_db}')
```

```
print(f'N betűk száma: {n_db}')
```

Magyarázat:

Az `utvonal.count('E')` megszámolja, hányszor van 'E' a szövegben.

A `print(f"...")` egy modern formázási mód (f-string), ahol a kapcsos zárójelek közé írt változó értéke jelenik meg a szövegben.

Itt azt mondjuk a Pythonnak: "Nézd meg az `utvonal` doboz tartalmát, és számold meg, hányszor szerepel benne az 'E' karakter."

Az eredményt (egy számot) elmentjük az `e_db` változóba. Ezt megcsináljuk mind a négy irányra (E, D, K, N).

3. Lépés: Az útvonal egyszerűsítése (Logika)

Ez a feladat legizgalmasabb része. Hogyan kapjuk meg a legrövidebb utat?

- Ha van 5 Észak (E) és 3 Dél (D) lépésünk, akkor a robot valójában csak **2-t ment Északra** ($5-3=2$).
- Ha a Dél lenne több (pl. 4 D és 1 E), akkor **3-at ment Délre** ($4-1=3$).
- Ugyanez igaz a Kelet (K) és Nyugat (N) irányokra.

Python

3. feladat: Útvonal egyszerűsítés

Észak-Dél tengely

if `e_db > d_db`:

`maradek_ed = "E" * (e_db - d_db)`

else:

`maradek_ed = "D" * (d_db - e_db)`

Kelet-Nyugat tengely

if `k_db > n_db`:

`maradek_kn = "K" * (k_db - n_db)`

else:

`maradek_kn = "N" * (n_db - k_db)`

Az eredmény kiírása

`print(f"Egy legrövidebb út parancsszava: {maradek_ed}{maradek_kn}")`

Magyarázat:

- Az if (ha) és else (különben) szerkezettel eldöntjük, melyik irányból van több.
 - A kivonással megkapjuk a nettó elmozdulást.
 - Pythonban a "E" * 3 eredménye "EEE", tehát egyszerűen megszorozhatjuk a karaktert a darabszámmal.
 - Végül a két irányt "összeadjuk" (egymás után fűzzük).

if e_db > d_db:: Megnézzük, több-e az Észak, mint a Dél.

Kivonás:

Ha igen, kivonjuk a kettőt egymásból (e_db - d_db). Ez megadja, hány lépés maradt "tisztán" Észak felé.

Szöveg sokszorozása:

Pythonban a szöveget lehet szorozni! Ha a kivonás eredménye 2, akkor az "E" * 2 eredménye "EE" lesz. Ezt elmentjük a maradek_ed változóba.

else:

Ha nem az Észak volt több (tehát a Dél több vagy egyenlő), akkor a Déli irányt számoljuk ki ugyanígy.

print():

Ez a függvény ír ki szöveget a képernyőre.

Az **f betű** az idézőjel előtt:

Ez jelzi a Pythonnak, hogy a szövegben kapcsos zárójelek { } lesznek.

{maradek_ed}:

A Python nem a betűket fogja kiírni, hanem "kinyitja a dobozt", és behelyettesíti az értékét (pl. "EE").

A teljes kód egyben

Másold be ezt egy Python fájlba (pl. robot.py), és futtasd le!

Python

1. feladat: Beolvasás

```
utvonal = input("Kérem a robot parancsait: ")
```

2. feladat: Számlálás és kiírás

```
e_db = utvonal.count('E')
```

```
d_db = utvonal.count('D')
```

```
k_db = utvonal.count('K')
```

```
n_db = utvonal.count('N')
```

```
print(f"E betűk száma: {e_db}")
```

```
print(f"D betűk száma: {d_db}")
```

```
print(f"K betűk száma: {k_db}")
```

```
print(f"N betűk száma: {n_db}")
```

3. feladat: Útvonal egyszerűsítése

Észak-Dél tengely

```
if e_db > d_db:
```

```
    maradek_ed = "E" * (e_db - d_db)
```

```
else:
```

```
    maradek_ed = "D" * (d_db - e_db)
```

Kelet-Nyugat tengely

```
if k_db > n_db:
```

```
    maradek_kn = "K" * (k_db - n_db)
```

```
else:
```

```
    maradek_kn = "N" * (n_db - k_db)
```

Az eredmény kiírása

```
print(f"Egy legrövidebb út parancsszava: {maradek_ed}{maradek_kn}")
```