

Informatika 2, 3. ZH (2020-05-19)

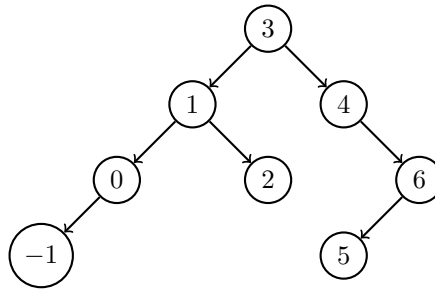
A megoldásokat a hazi@math.bme.hu címre küldjétek el 19:15-ig!

1. Bővítsük ki az órán tanult `Tree` nevű osztályt, ami rendezett bináris fát reprezentál, egy `edges` metódussal, ami összeszámolja a fa élét. (5 pont)

Példa:

```
t = Tree(3)
for i in [1, 2, 0, -1, 4, 6, 5]:
    t.insert(i)
print(t.edges())
```

Eredmény: 7



1. ábra. Ezek a csúcsok lettek sorban beillesztve: [3,1,2,0,-1,4,6,5]

Segítség: 11. gyakorlat.

2. Tekintsük az alábbi háromszög alakú négyzetrácsot:

1			
1	1		
1	2	2	
1	3	5	5

2. ábra. Hányféle képen lehet a felső cellából az adott cellába eljutni, ha csak lefelé és jobbra léphetek?

Írjunk olyan függvényt, ami megmondja, hogy a bal felső cellából hányféle képen lehet a többibe eljutni, ha csak lefelé vagy jobbra léphetünk. Az ábrán a megoldás bele van írva a cellákba. (5 pont)

- A függvény neve legyen `lepes`.
- Egy paramétere legyen, egy pozitív egész szám, hogy mekkora háromszögre akarjuk kiszámolni az eredményt.
- A visszatérési értéke listák listája legyen, ami a kitöltött háromszög sorait tartalmazza. A példában:

`lepes(4) → [[1], [1, 1], [1, 2, 2], [1, 3, 5, 5]]`

Segítség: a binomiális együtthatókhöz hasonlóan itt is minden cella a felette és tőle balra lévő összege, kivéve a háromszög szélein.

3. Írjunk olyan függvényt, amit egy zárójeleket tartalmazó sztringben `$`-re cseréli a zárójelben lévő részeket, de kétféle zárójel lehet: `()` és `{}`. Nekünk azokat a részeket kell lecserélni, melyek legkülső szintű zárójelek, mindegy hogy melyik fajtából. (5 pont)

- A függvény neve legyen `zarojel2`.
- Egy paramétere legyen, egy sztring, ebben egy két zárójellel jól-zárójelezett kifejezés lesz.
- A visszatérési értéke legyen egy ugyanolyan hosszú sztring, ahol a zárójelezett részek ki vannak cserélve.

Például:

`zarojel2("1/(2+(-3))-{4+(-5)}") → "1/$$$$$$$$-$$$$$$$$"`

4. Írjunk olyan függvényt, ami három dimenziós egységvektorokat generál véletlenszerűen, használjunk `numpy`-t. (5 pont)

- A függvény neve legyen `unit3`.
- Egy paramétere legyen: n , egy pozitív egész szám, hogy hány vektort szeretnénk.
- A visszatérési értéke legyen egy $\mathbb{R}^{n \times 3}$ mátrix, ahol minden sor egy három dimenziós egységvektor.

Például:

```
>>> unit3(2)
array([[ -0.74790227, -0.47255394,  0.46619199],
       [ 0.19888237, -0.69017507,  0.69577595]])
```

Segítség: generáljunk véletlen számokat a $[-1, 1]$ intervallumban, olyan méretűt mint a kívánt mátrix. Majd soronként osszuk el őket a saját normájukkal.