

Tartalomjegyzék

- 1 4. gyakorlat - opcionális paraméter és könyvtárak
 - ◆ 1.1 Feladatok
 - ◆ 1.2 Feladatok megoldása
 - ◇ 1.2.1 buli
 - ◇ 1.2.2 polarbol_sima
 - ◇ 1.2.3 matrixlog
 - ◇ 1.2.4 huto
 - ◆ 1.3 Házi feladatok megoldása
 - ◇ 1.3.1 cegesbuli
 - ◇ 1.3.2 teki

4. gyakorlat - opcionális paraméter és könyvtárak

Ezen a gyakorlaton az el?z? heti el?adás anyagát dolgozzuk fel.

Feladatok

Feladatok a CloudCoder-en megtalálhatóak. Ajánlott sorrend:

1. buli
2. polarbol_sima
3. matrixlog
4. huto

Feladatok megoldása

buli

Megoldás listaértelmezés nélkül:

```
import calendar

def buli(napok, honap, ev=2015):
    hetvegek = []
    for nap in napok:
        if calendar.weekday(ev, honap, nap) > 4:
            hetvegek.append(True)
        else:
            hetvegek.append(False)
    return hetvegek
```

Megoldás listaértelmezéssel:

```
import calendar

def buli(napok, honap, ev=2015):
    return [calendar.weekday(ev, honap, nap) > 4 for nap in napok]
```

polarbol_sima

Megoldás listaértelmezéssel:

```
import math

def polarbol_sima(polarok):
    return [(round(hossz*math.cos(szog)), round(hossz*math.sin(szog))) for hossz, szog in polarok]
```

matrixlog

Egy olyan megoldás ami nem módosítja a paraméterként kapott *mutable* listákat:

```
import math

def matrixlog(n, M):
    if n == 0:
        uj_szam = round(math.log(M, 2), 2)
        return uj_szam
    else:
        uj_M = []
        for elem in M:
            uj_elem = matrixlog(n-1, elem)
            uj_M.append(uj_elem)
        return uj_M
```

huto

```
import calendar

def huto(ev, honap, felir):
    honaphossz = calendar.monthrange(ev, honap)[1]

    return [szam - (honaphossz - nap) for nap, szam in felir]
```

Házi feladatok megoldása

cegesbuli

```
import calendar

def cegesbuli(ev, honap, nap=1):
    temak = [
        "Hepi Hetfo",
        "Kellemes Kedd",
        "Szerelmes Szerda",
        "Csaloka Csutortok",
        "Perverz Pentek",
        "Szomoru Szombat",
        "Vidam Vasarnap"]

    return temak[calendar.weekday(ev, honap, nap)]
```

teki

```
import math

def teki(k, lepeshossz, fi):
```

polarbol_sima

```
x, y = 0, 0
szog = 0
for i in range(k):
    x = x + lepesossz*math.cos(szog * math.pi / 180)
    y = y + lepesossz*math.sin(szog * math.pi / 180)
    szog += fi
return (round(x), round(y))
```