

veliki-kih

January 28, 2024

0.1 Veliki kih

Neka visokomogočna in precej velika bitja, so zbrala 65536 zvezd najrazličnejših svetlosti. Razporedila so jih na dvodimenzionalno mrežo velikosti 256x256. Če so stvaritev pogledala od daleč, je bila videti kot črno-bela slika. (No, v resnici črno-oranžno-rumeno-bela, saj gre za zvezde). Zvezde so torej predstavljale piksele neke oranženkaste slike.

Slika je bila imenitna in znana in imela celo ime. Rekli so ji ... bogvekakoli, tudi ime se je izgubilo.

Dolga leta je bila razstavljena. Potem je eden od ogledovalcev kihnil. Močno kihnil.

Zvezde so se razletele vsaka na svojo stran. Ene počasneje, ene hitreje; ene levo, ene desno, ene gor, ene dol; večina, pač, nekam postrani.

Okrog zvezd so se sčasoma naredili planeti. Na planetih je zrasel najprej en sam plevel, kasneje pa tudi solata in žirafe. In neka manj mogočna in manj velika bitja, ki pa so sčasoma odkrila teleskope in spektre in tako naprej. V datoteko `meritve.txt` so zbrala podatke o meritvah. Vsaka vrstica datoteke predstavlja eno zvedo: prvi stolpec pove njeno svetlost, drugi in tretji njeni koordinati, četrti in peti pa njeno hitrost v smeri x in smeri y (torej: za koliko se spremenijo koordinate zvezde v enem stoletju). Ker so ona, prva bitja, bila in tudi ostala zelo diskretna, so vsa števila cela.

Kako je bilo ime sliki?

Namig: za začetek bo treba ugotoviti, kaj je bilo na njej ... pred velikim kihom.

0.1.1 Tehnična pomoč

Izziva se lahko seveda lotiš v poljubnem jeziku. Ker se na predavanjih pogovarjamo o numpyju in Pythonu, pa je tule pomoč za numpy in Python.

Podatke najlažje preberemo z

```
[1]: import numpy as np

np.loadtxt("meritve.txt", dtype=np.int64)

[1]: array([[ 161, -73865,  37242,   -6,    3],
 [ 161, -61517, -110923,   -5,   -9],
 [ 157, -123325,  98876,  -10,    8],
 ...,
 [ 104, -12324,  12370,   -1,    1],
 [ 104,  98860,  86456,    8,    7],
```

```
[ 108, 74256, 12459, 6, 1]])
```

Kako to premetavati naprej, odkrij sam(a).

Če želim v jupyteru pokazati sliko dimenzij (30, 50), ki ima neko bolj temno zvezdo na (5, 7) in neko svetlejšo na (25, 33), nam je storiti tole:

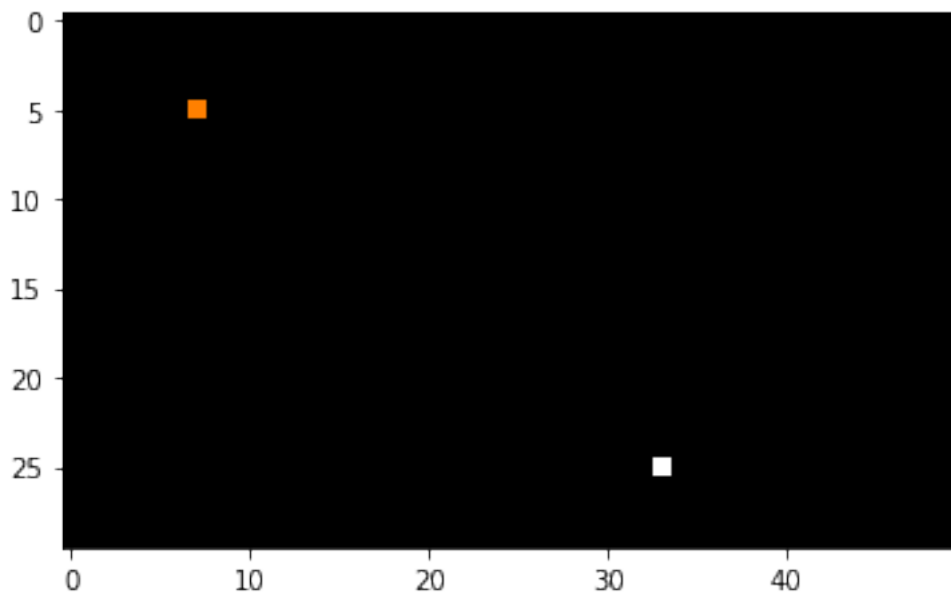
```
[2]: %matplotlib inline

import numpy as np
import matplotlib.pyplot as plt

slika = np.zeros((30, 50), dtype=np.uint8)
slika[5, 7] = 64
slika[25, 33] = 128

plt.imshow(slika, cmap="afmhot")
```

```
[2]: <matplotlib.image.AxesImage at 0x7fb8de199550>
```



Z `dtype=np.uint8` v klicu `np.zeros` povemo, da matrika vsebuje cela števila (osembitne nepredznačene int-e). V tem primeru `plt.imshow` pričakuje vrednosti med 0 in 255, kar ravno ustreza svetlostim zvezd. (Kakšno naključje.)

Argument `cmap="afmhot"` v klicu `plt.imshow` določi barvno paletu. Če bi napisali `cmap="gray"`, bi bila slika črno-bela. Ker gre za sliko iz zvezd, bo ustrezna paleta "afmhot".

Namig: ni težko, samo simulacijo je potrebno pognati nazaj. Zadnja dva stolpca povesta, za koliko

se drugi in tretji stolpci spremenita v vsakem stoletju. Odkriti morate, za koliko stoletij **nazaj** je potrebno iti, da se bodo vse številke v drugem stolpcu razlikovale za največ 256; v drugem pa ravno tako. Ko dobite takšne koordinate, pa izrišite sliko po gornjih navodilih.