

Class EPS 109 "Computer Simulations in Earth and Planetary Science"

Computer lab assignment 2

Instructor: Burkhard Militzer

```
In [2]: #load the libraries as usual
import matplotlib.pyplot as plt
import numpy as np
```

(1) Type in the following commands, see what happens and interpret them

```
In [ ]: data = np.random.random(size=(4, 6))
plt.imshow(data, interpolation='nearest')
plt.show()
print(data)
```

```
In [ ]: data = np.random.random(size=(11, 11))
data2 = np.round(data)
plt.imshow(data2, interpolation='nearest')
plt.show()
print(data2)
```

```
In [ ]: n = 11
data = np.zeros((n,n))

# (a) what is the effect of the following three lines
data[1,1] = 1
data[2,3] = 1
data[3,5] = 1
# (b) again what do the following two lines do?
data[5,1:n-1] = 1
data[n-2,:] = 1

print(data)
plt.imshow(data, interpolation='nearest')
plt.show()
```

```
In [ ]: # Here we practise 'for' loops
n = 11
data = np.zeros((n,n))

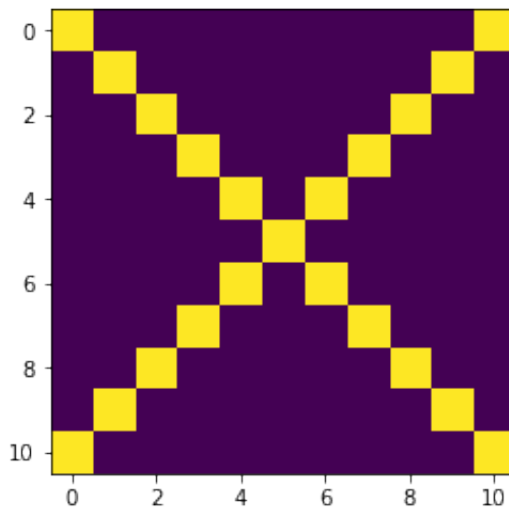
# (a) What is the effect of this 'for' loop?
for i in range(0,n):
    data[i,3] = 1

# (b) And this 'for' loop?
col = (n-1) // 2
for i in range(0,n,2):
    data[i,col] = 1

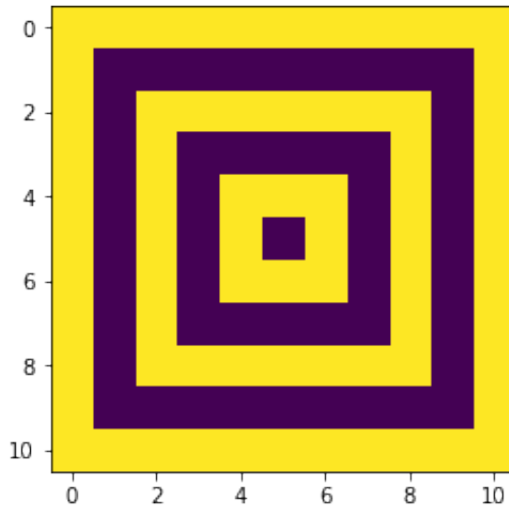
print(data)
plt.imshow(data, interpolation='nearest')
plt.show()
```

(2) Let us switch gears. You are asked to reproduce the following images by replacing the ... dots with your own Python commands. Take advantage of the examples above.

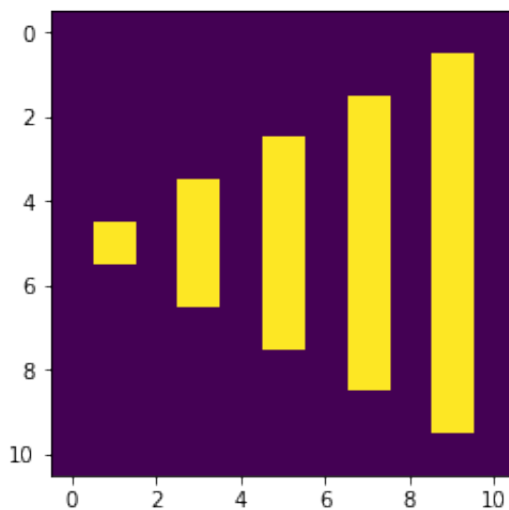
```
In [3]: n = 11
data = np.zeros((n,n))
...
print(data)
plt.imshow(data, interpolation='nearest')
plt.show()
```



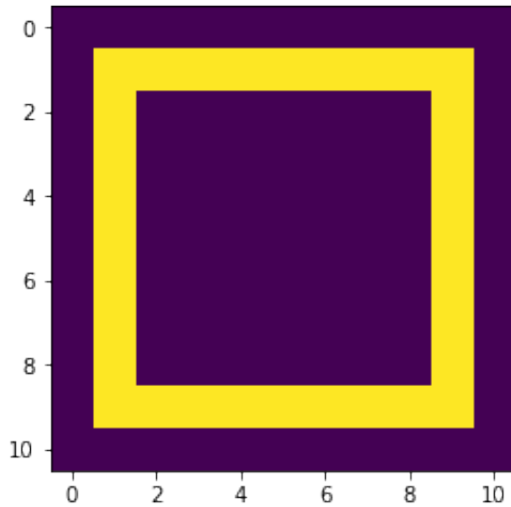
```
In [4]: n = 11
data = np.zeros((n,n))
...
print(data)
plt.imshow(data, interpolation='nearest')
plt.show()
```



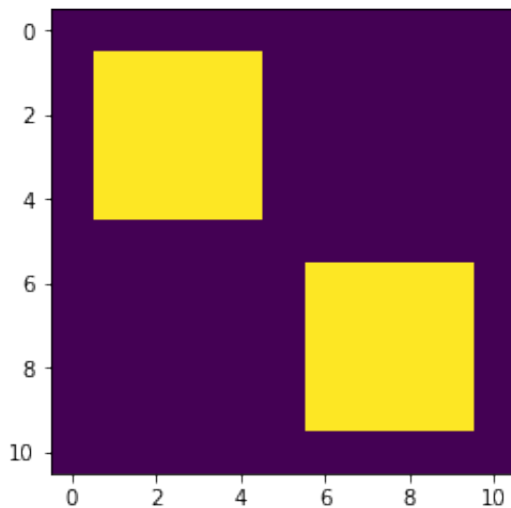
```
In [36]: n = 11
data = np.zeros((n,n))
...
plt.imshow(data, interpolation='nearest')
plt.show()
```



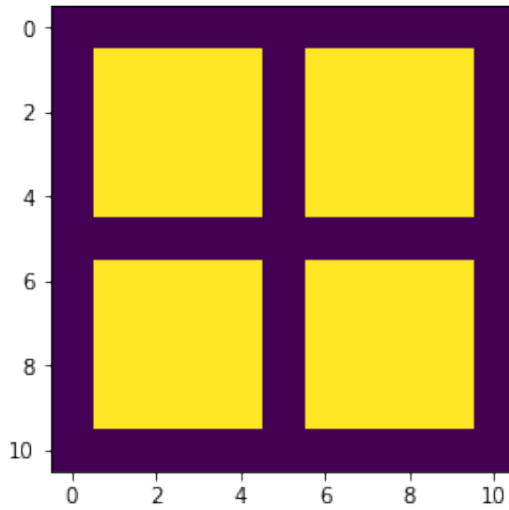
```
In [37]: n = 11  
data = np.zeros((n,n))  
...  
plt.imshow(data, interpolation='nearest')  
plt.show()
```



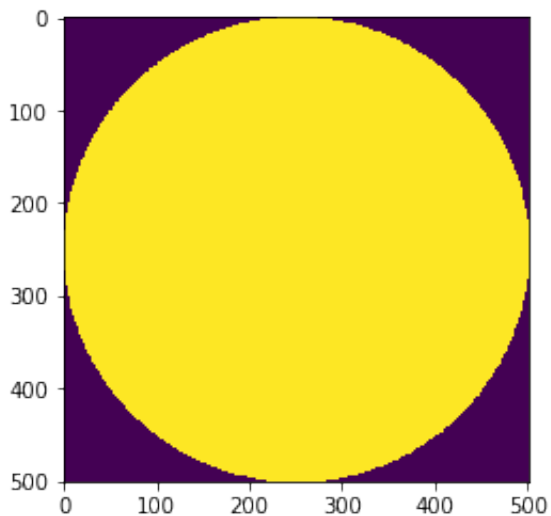
```
In [38]: n = 11  
data = np.zeros((n,n))  
...  
plt.imshow(data, interpolation='nearest')  
plt.show()
```



```
In [5]: n = 11  
data = np.zeros((n,n))  
...  
print(data)  
plt.imshow(data, interpolation='nearest')  
plt.show()
```



```
In [39]: n = 501  
data = np.zeros((n,n))  
...  
plt.imshow(data, interpolation='nearest')  
plt.show()
```



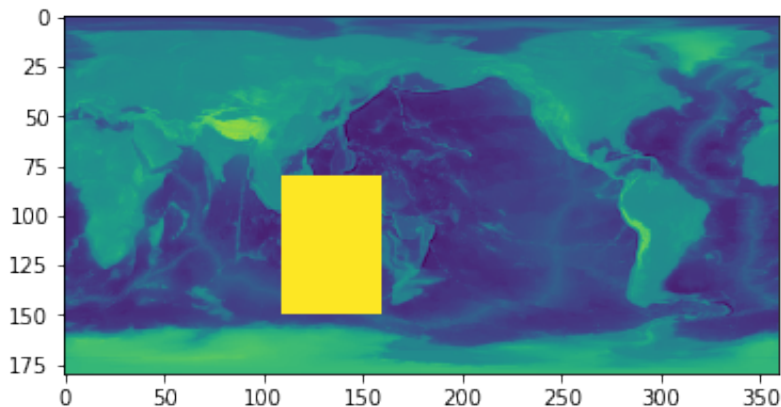
(3) Finally we want to load and display an external data set that contains Earth's elevation and bathymetry data of grid of longitude and latitude points

```
In [ ]: #The following command will fail if you do not have the 'topography_180x360_grid.txt' file in current folder
H = np.loadtxt('topography_180x360_grid.txt')
print(H)
print(H.shape)
```

```
In [ ]: # Now you should simply be in awe
plt.imshow(H, interpolation='nearest')
plt.show()
```

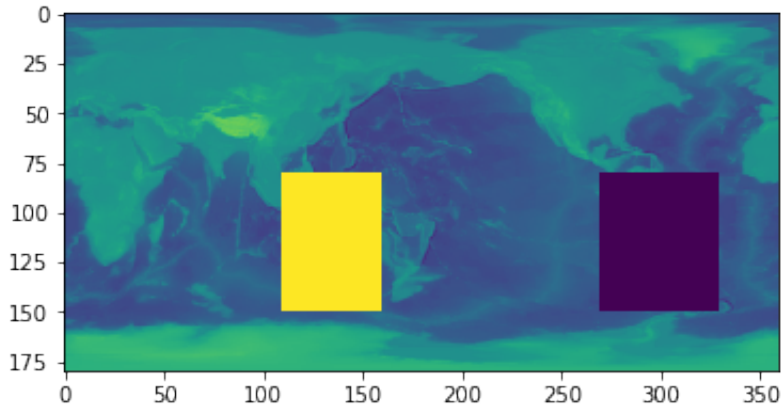
(3a) Insert one commnd and cover the continent of Australia with a yellow rectangle. Set the values in the rectangle to -10000.

```
In [15]: H2 = H
...
plt.imshow(H2, interpolation='nearest')
plt.show()
```



(3b) Insert one commnd and cover the continent of South American with a blue rectangle. Set the values in the rectangle to -10000.

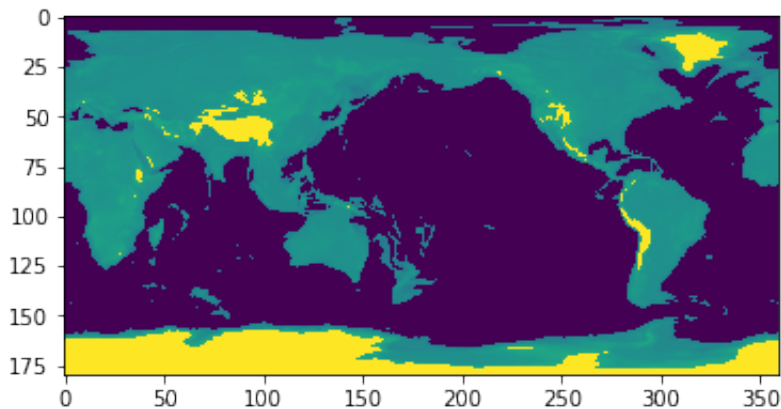
```
In [16]: H3 = H
...
plt.imshow(H3, interpolation='nearest')
plt.show()
```



```
In [ ]: #Try the following commands. The result will be very odd. What happened to the H array?
H4 = H
plt.imshow(H4, interpolation='nearest')
plt.show()
```

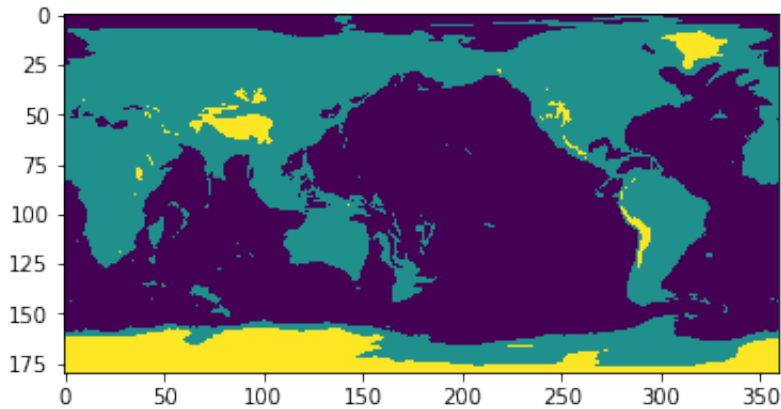
(3c) Load the 'topography_180x360_grid.txt' file again. For all points with elevations above 2000 m, set the elevation to 10000 m. For all points with a pessimetry of more then 2000 m, set the elevation to -10000 m.

```
In [18]: H4 = np.loadtxt('topography_180x360_grid.txt')
...
plt.imshow(H4, interpolation='nearest')
plt.show()
```



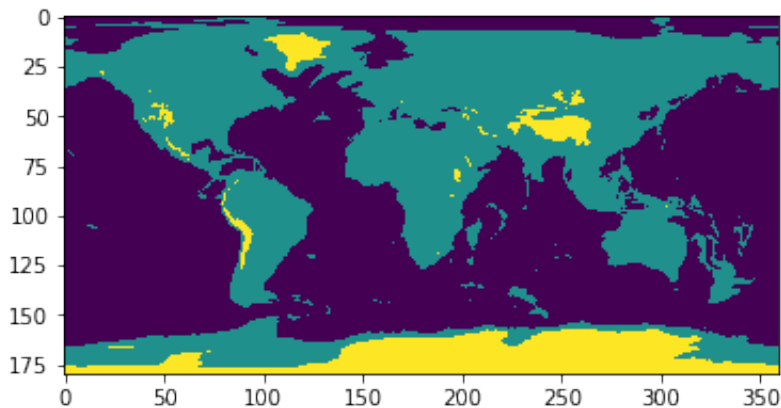
(3d) In addition to the modifications in (3c), now also change the elevation to zero if the original elevation was between -2000 and $+2000$ m.

```
In [19]: H4 = np.loadtxt('topography_180x360_grid.txt')
...
plt.imshow(H4, interpolation='nearest')
plt.show()
```



(3e) From the movie "Black Panther", we learned that Wakanda is the center of the modern world. Now shift our map in east-west direction so that Africa appears in the map's center as shown below.

```
In [20]: #Accommodating Wakanda
H5 = np.copy(H4);
...
plt.imshow(H5, interpolation='nearest')
plt.show()
```



(3f) Let us practise the np.append command. Just type in the commands to see what happens.

```
In [ ]: H = np.loadtxt('topography_180x360_grid.txt')
        HH = np.append(H,H,axis=0)
        plt.imshow(HH, interpolation='nearest')
        plt.show()
```

```
In [ ]: H = np.loadtxt('topography_180x360_grid.txt')
        HH = np.append(H,H,axis=1)
        HHHH = np.append(HH,HH,axis=1)
        plt.imshow(HHHH, interpolation='nearest')
        plt.show()
```