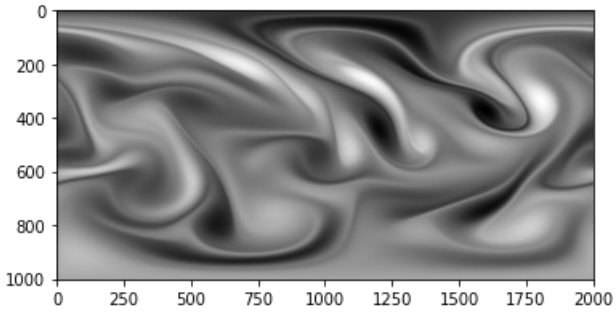
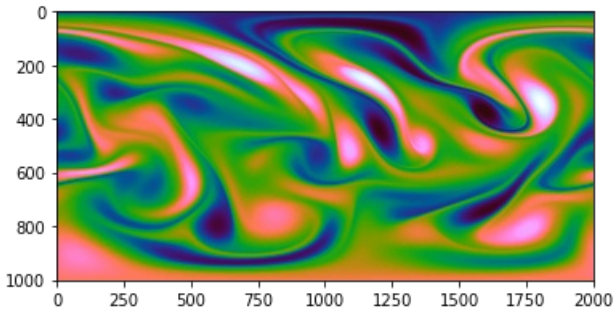
In [25]:

```
#Task1(A)

import matplotlib.pyplot as plt
import matplotlib.image as mpimg

#Load the image file as an array
img1 = mpimg.imread('vorticity.png')

#Display it using the "imshow" command of Matplotlib
#Use the "gray" colormap from Matplotlib
imgplot1 = plt.imshow(img1, cmap='gray')
```



In [26]:

```
#Task1(B)

import matplotlib.pyplot as plt
import matplotlib.colors as nc
import numpy as np
import matplotlib.image as mpimg

colors = np.loadtxt('mycolormap.txt')
mycmap = nc.ListedColormap(colors, N=None)

#Save the image as an PNG-file with name "vorticity_with_cmap.png"
#using my own colormap as created in Exercise: your own colormap
plt.imsave('vorticity_with_cmap.png', img1, cmap=mycmap)

#loading it in the notebook.
img2=mpimg.imread('vorticity_with_cmap.png')

#Displaying it in the notebook.
imgplot2 = plt.imshow(img2)
```



In [27]:

```
#Task 1(C)

import os
#Compare the exact file-size of the original image file "vorticity.png" and
"vorticity_with_cmap.png".
a = os.path.getsize('vorticity.png')
b = os.path.getsize('vorticity_with_cmap.png')
```

```
display(a)
display(b)

#shape of the corresponding image arrays.
display(img1.shape)
display(img2.shape)
```

```
301322
```

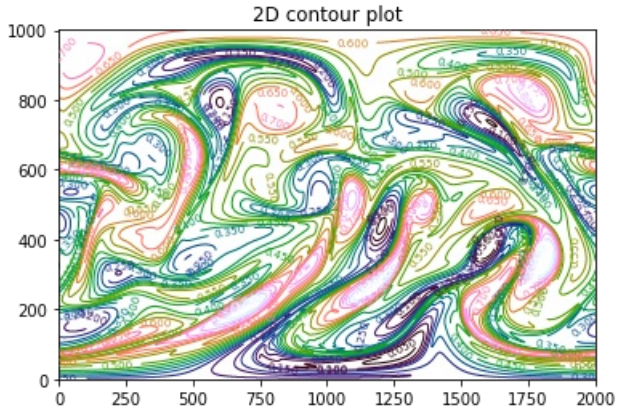
```
438776
```

```
(1002, 2004)
```

```
(1002, 2004, 4)
```

In [28]:

```
#Task 1(D)

import matplotlib.pyplot as plt
import matplotlib.colors as nc
import numpy as np

colors=np.loadtxt('mycolormap.txt')
mycmap=nc.ListedColormap(colors, N=None)

#Make a plot of the level-sets of "vorticity.png" using the Matplotlib command "contour"
#Use 20 levels.
#Color the level-set lines by my own colormap.
C = plt.contour(img1, 20, cmap=mycmap, linewidths = 1)
L = plt.clabel(C, inline=1, fontsize=7)
T = plt.title('2D contour plot')
plt.show()
```



In [ ]:

In [ ]:

In [ ]: