Assignment 2 (Due Friday 20th September):

Task: Use your code from Lab 3 to create a python routine to find a 2D bracket of a 2-variable function.

Details:

You should create a function **minbracket2D** that can be called in the following way (see **testfile1.py**)

```
 8  import numpy as np
 9  import mysearch as mys
10
11  def f(x,y):
12     return (x-0.2)**2+(y-0.3)**2
13
14  bregion = [[-2.0,2.0],[-3.0,3.0]]
15  N = 20
16
17  blist = mys.minbracket2D(f,bregion,N)
18
19  print(blist)
```

The function **minbracket2D** should therefore be contained within your own **mysearch.py** module.

The inputs to **minbracket2D** are:

f(x,y):	A handle to a two-variable function
bregion:	An array of floats [[ax,bx],[ay,by]] to look for the brackets in
N:	An integer number of points to do the bracketing.

The output to **minbracket2D** should be a list of coordinates in x and y that bracket the minima (see **output1.txt**). E.g.

```
In [4]: run testfile1.py
[[[0.0, 0.40000000000000036], [0.0, 0.5999999999999996]]]
```

If no minima are found the list should be empty.

Important notes:

The module must work on the first try, and for an arbitrary function as an input. If it does not then marks will be deducted.

Try to make you code as robust as possible so that it does not miss any zeros. You may have to modify your bracketing procedure to do this.

Files to submit:

You should upload (only) the following files:

mysearch.py	- a python module that contains the **minbracket2D** function, as well as any other functions that it needs to run
Lab3_Q2.py	- the code from Lab 3 that you used/adapted to create your bracketing function.

This can be your own code or from the provided solutions.

Grading:

The code will be graded according to the following scale:

Compliance: This is whether the code is submitted as instructed	30%

Effectiveness (i.e. whether the code passes independent tests):	50%
Comments (whether they are comprehensible):	20%

Important information (academic integrity):

1. You have to write the code yourself. Any copying of code from an external source other than the lab solutions that you provide will result in a mark of zero being awarded.

2. **The code must be clearly adapted from your Lab solutions.** If the code does not do this then a mark of zero may be awarded. If your code from the Labs does not work then you can use the Lab Solutions provided on Canvas (include these files in the upload instead of your own).