

PHYS-GA 2000 Computational Physics PS2

Kaixuan Zhou

September 19, 2023

Prob 1:

```
32bit for the given value is 01000010110010011111100110101011
exponet= 10000101
mantissa= 10010011111100110101011
The difference of the two expression is 2.7514648479609605e-06
```

Figure 1: This is the result of given question1

Prob 2:

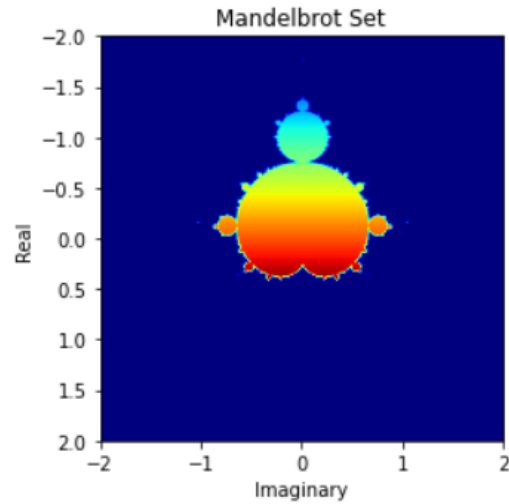
We used two ways solving the problem. The two results are satisfying of

```
-1.7418198158396654
-1.7418198158396654
-1.7418198158396654
-1.7418198158396654
-1.7418198158396654
-1.7418198158396654
-1.7418198158396654
13 s  $\pm$  307 ms per loop (mean  $\pm$  std. dev. of 7 runs, 1 loop each)
```

```
-1.7418198158362388
-1.7418198158362388
-1.7418198158362388
-1.7418198158362388
-1.7418198158362388
-1.7418198158362388
-1.7418198158362388
362 ms  $\pm$  37.1 ms per loop (mean  $\pm$  std. dev. of 7 runs, 1 loop each)
```

the accuracy. The first one with for loop has 13s running time average, the one without for loop only has 362ms running time average. Meshgrid works much better here with 100 iterations.

Prob 3:
This is the figure for the Mandelbrot set.



Prob 4: This is the result of given function with two different ways of solving

```
In [1]: runfile('C:/Users/kz761/Documents/GitHub/phys-ga2000/ps-2/prob4_1and2.py', wdir='C:/Users/kz761/Documents/GitHub/phys-ga2000/ps-2')
(-9.999894245993346e-07, -999999.999999)
(-1.000000000001e-06, -1000010.5755125057)
```

it. We can see there's some difference on the accuracy. So I merged them and passed the pytest given in the problem set.

GitHub account:luminousxuan