Lab 6

Exercise 1

- Code for our Jacobi function is inside of lib.py
- For b as [1,1,1], we get our x as [0.09018568 0.0265252 0.02387268], and we know this works since A * x does equal our b value.

Exercise 2

- Code for our Gauss Seidel is inside of lib.py
- For b as [1,1,1], we get our x as [0.09018568 0.0265252 0.02387268], and we know this works since we got the exact same thing for our previous solution.

Exercise 3

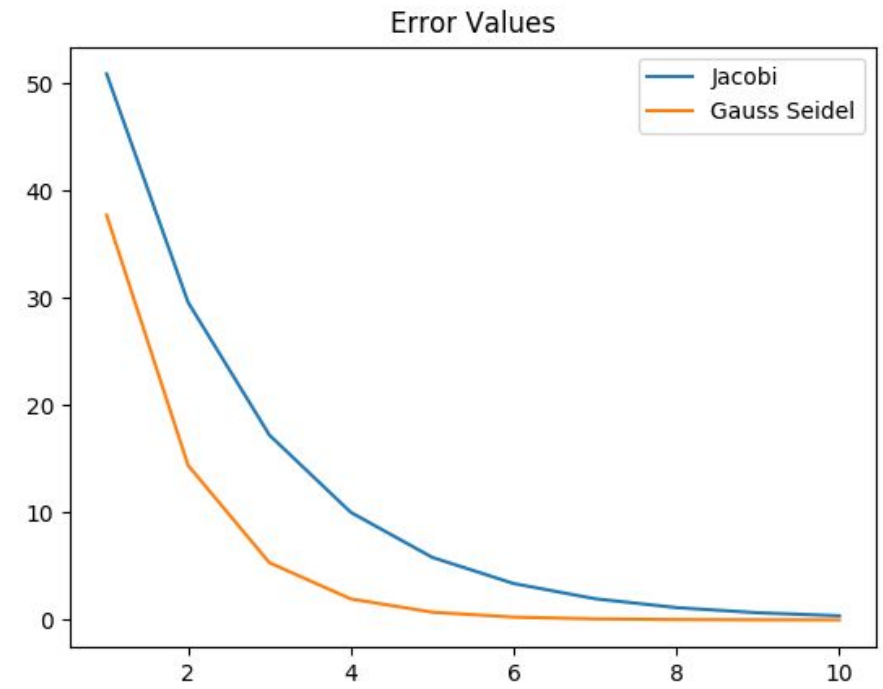
- Jacobi and Gauss Seidel got x as [1. -2. 3. 1. 5.]. Jacobi needed total of 42 iterations and Gauss Seidel needed a total of 22 iterations.

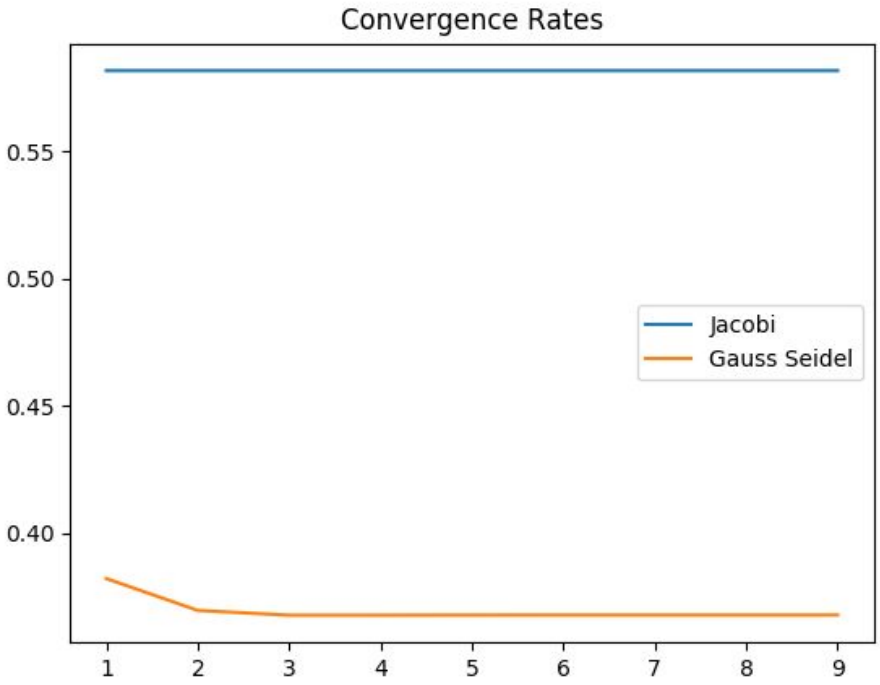
Exercise 4

- Jacobi and Gauss Seidel both got the final solution of:

```
[0.21132487 0.15470054 0.16987298 0.16580754 0.16689687 0.16660498
 0.16668319 0.16666224 0.16666785 0.16666635 0.16666675 0.16666664
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667 0.16666667
 0.16666667 0.16666667 0.16666667 0.16666667 0.16666664 0.16666675
 0.16666635 0.16666785 0.16666224 0.16668319 0.16660498 0.16689687
 0.16580754 0.16987298 0.15470054 0.21132487]
```

Exercise 5:


Error Values
Jacobi
Gauss Seidel
50
40
30
20
10
0
2
4
6
8
10


Convergence Rates
Jacobi
Gauss Seidel
0.55
0.50
0.45
0.40
1
2
3
4
5
6
7
8
9