Homework 3

- Due 11/11 23:59 pm est
- Following instructions provided in **Homework submission instructions**

Problem 1 (15 points)

part (a)

For the function define below, calculate the exact number of atomic operations that are needed as function of n (where n is the length of input list L). Please give explainations for every term.

In [ ]:
```
def f(L):
    newlist = []
    for i in L:
        if i % 2 == 0:
            newlist.append(i)
    return newlist
```

Hint:

1. Creating a empty list costs 1 atomic operation.
2. Focus on the worst case scenario.

part (b)

What is the running time if use Big-O notation?

Problem 2 (25 points)

part (a)

To check if there is any duplicates in a list, we have covered 3 different solutions. Can you come up with a solution that is more efficient than them?

In [ ]:
```
```

part (b)

For `n = [800, 1600, 3200, 6400, 12800]`, use the timetrials function we defined in the notebook, calculate the running time for **duplicates2** and your function to verify that your function is indeed more efficient.

In [ ]:

```

```

Problem 3 (30 points)

The idea of Fixed point algorithm and its procedures are given in the note. Write a function to implement the Fixed-point algorithm.

In [ ]:

```

```

Problem 4 (30 points)

The Cauchy(θ,1) distribution has the following probability density function

$$f(x \mid \theta) = \frac{1}{\pi[1 + (x - \theta)^2]},$$

where θ is the unknown parameter.

Suppose x_1, x_2, ..., x_n are observed independent sample from the Cauchy distribution, then the loglikelihood function for the data can be written as:

$$\ell(\theta) = -n \log \pi - \sum_{i=1}^{n} \log(1 + (x_i - \theta)^2).$$

The maximum likelihood estimator (mle) for θ, denoted as θ_0, is then the solution of the following equation

$$f(\theta) = \frac{d\ell(\theta)}{d\theta} = -2 \sum_{i=1}^{n} \frac{\theta - x_i}{1 + (\theta - x_i)^2} = 0.$$

Then, the root finding techniques we have learned can be used here. Essentially, we are looking for θ_0 such that $f(\theta_0) = 0$.

Part (a)

Now, suppose our observed data x_i's are defined in a list called data such that:

In [1]:

```
data = [-2.44, 3.29, 3.71, -2.40, 4.53, -0.07, -1.05, -13.87, -2.53,
        -1.75, 3.30, 1.77, -0.23, 2.76, 3.80, 3.47, 56.75, -1.34, 4.24]
```

Given the data, the log-likelihood function can be shown in the following plot. Use the

newton method you wrote during the class, find the maximizer for the function below.

loglikelihood

Hint:

1. The newton method requires the derivate of function f, f'. The expression for f' is given below:

$$f'(\theta) = -2\sum_{i=1}^{n} \frac{1-(\theta - x_i)^2}{[1+(\theta - x_i)^2]^2}$$

1. Given the x_i's in list called data, if you want to write a function such that

$$g(\theta) = \sum_{i=1}^{n} (\theta - x_i)^2,$$

one possible way is

```
def g(theta):
    return(sum([(theta-x)**2 for x in data]))
```

1. SET **init = 4** and set **max_iter = 100**.

In [ ]:

Part (b)

Use the **Fixed point** method you wrote in problem 3, find the root for the function above.

Hint:

1. set **alpha = 0.25**.
2. set **init = 4**.
3. set **max_iter = 100**.

In [ ]: