

Homework 1:

Problem 1:

part 1:

```
def Fib_loop(n=0):  
    m1 = 1  
    m2 = 1  
    m3 = m1+m2  
    if n==1 or n==2:  
        return 1  
    for i in range(0,n-2):  
        m3 = m2+m1  
        m1=m2  
        m2=m3  
    return m3
```

part 2:

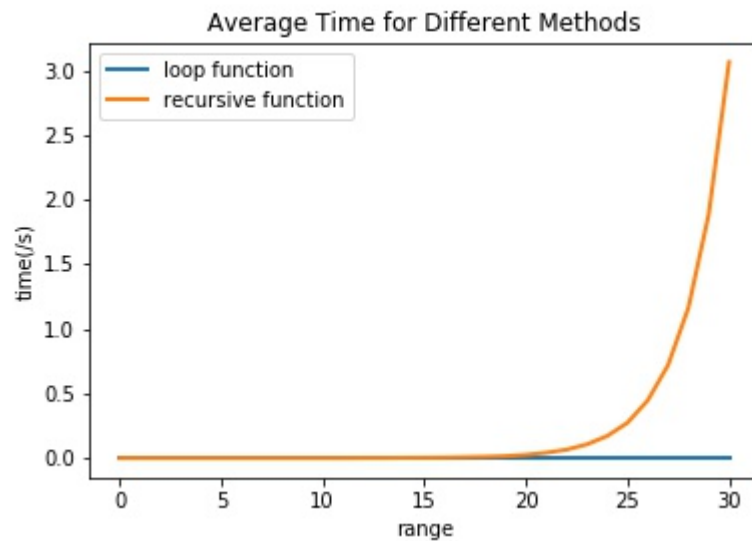
Compared with the recursive one, function which use loop will run faster because for recursive, it must calculate it floor by floor and it will cost a lot of time to create each floor then go back to the first floor. For example, it is just like go upstairs to the highest floor and then go down to the ground to get the result but for function which use loop, it just calculate it step by step and then give the result directly. Also, the recursive one will cost much more memory compared with the loop one.

part 3:

```
import time
def Fib_rec(n=0):
    if n==0:
        return 1
    if n==1:
        return 1
    else:
        return Fib_rec(n-1) + Fib_rec(n-2)
def timer(k,n=0,f=Fib_rec):
    sum = 0
    for i in range(0,k):
        x = time.time()
        f(n)
        x = time.time() - x
```

```
sum += x  
average = sum/k  
print(average)()
```

part 4:



Problem 2:

part 1:

$$M = \begin{pmatrix} 1 & -4 & 2 \\ -4 & 1 & -2 \\ 2 & -2 & -2 \end{pmatrix}$$

$$\det(M) = 1 \times (-2 - 4) - (-4) \times (8 + 4) + 2 \times (8 - 2) \\ = -6 + 48 + 12 = 54 \neq 0 \Rightarrow \text{inverse exist}$$

$$M^T = \begin{pmatrix} 1 & -4 & 2 \\ -4 & 1 & -2 \\ 2 & -2 & -2 \end{pmatrix}$$

$$M^{-1} \cdot M = I \Rightarrow M^{-1} = \begin{pmatrix} -\frac{1}{9} & -\frac{2}{9} & \frac{1}{9} \\ -\frac{2}{9} & -\frac{1}{9} & -\frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{5}{18} \end{pmatrix}$$

part 2: $M \cdot X = \lambda X$

$$\Rightarrow \begin{vmatrix} 1-\lambda & -4 & 2 \\ -4 & 1-\lambda & -2 \\ 2 & -2 & -2-\lambda \end{vmatrix} = 0$$

$$\Rightarrow (1-\lambda)((\lambda-1)(2+\lambda)-4) + 4(8+4\lambda+4) + 2(8-2+\lambda) = 0$$

$$\Rightarrow \lambda = -3, 6$$

$$\Rightarrow X_1 = \begin{pmatrix} -0.75 \\ -0.67 \\ 0.60 \end{pmatrix} \quad X_2 = \begin{pmatrix} -0.60 \\ 0.67 \\ 0.52 \end{pmatrix} \quad X_3 = \begin{pmatrix} 0.30 \\ -0.33 \\ 0.85 \end{pmatrix}$$

Part 3:

$$\nabla_A f(A) = \begin{pmatrix} 2X_{11}X_{22}X_{23} + X_{12}X_{13}X_{31} & X_{11}X_{13}X_{31} & X_{11}X_{12}X_{31} \\ -X_{11}^2X_{12} & X_{11}^2X_{23} & \sqrt{2}X_{11}X_{12} \end{pmatrix}$$

$$\nabla_A f(A) = \begin{pmatrix} -x_{11}^2 x_{32} & x_{11}^2 x_{23} & x_{11}^2 x_{22} \\ x_{11} x_{12} x_{13} & -x_{33}^2 x_{21} & -2x_{11} x_{12} x_{21} \end{pmatrix}$$

Part 4:

$$\frac{\partial g}{\partial x} = 3x^2 y + yz \cos(x) + y^2 z^5 \quad \frac{\partial g}{\partial y} = x^3 + z \sinh(x) + 2x y z^5$$

$$\frac{\partial g}{\partial z} = y \sinh(x) + 5x y^2 z^4$$

$$H(g) = \begin{pmatrix} 6xy - yz \sinh(x) & 3x^2 + z \cos(x) + 2yz^5 & y \cos(x) + 5y^2 z^4 \\ 3x^2 + z \cos(x) + 2yz^5 & 2x z^5 & \sinh(x) + 10xy z^4 \\ y \cos(x) + 5y^2 z^4 & \sinh(x) + 10xy z^4 & 20xy^2 z^3 \end{pmatrix}$$

Problem 3:

part 1:

The validation samples are used to tune the parameters of the algorithm and the test samples are used to test the algorithm. Which means, the validation samples are used to create the algorithm while the test samples are used to value the algorithm whether useful or not.

part 2:

Supervised learning algorithm: What we have is a lot of labeled data and what we want to do is to make some predictions. For example, we have some classified images and we also have some unclassified images so we can use supervised learning algorithm to classify those images.

Unsupervised learning algorithm: We don't have any data with label so the algorithm need to learn how to label those data by it self. For example, we have some news which we don't know their classification, so we need the unsupervised learning algorithm to help us to put them into different class.

The most important difference is whether the input data with label or not. If input is labeled data, it should be supervised learning algorithm, or not, unsupervised learning algorithm.