设计思路:$AX = B$

外循环使得矩阵成为右上矩阵,根据最大元交换的条件生成初等矩阵P_i, Q_i,新增广矩阵为$P_iAQ_i|P_iB$,记录 $Q* = Q_i, Q_0 = E$

内循环使得矩阵成为对角矩阵

最后循环一次执行消去操作即可

Finally

$$(x_1, x_2, \cdots x_n)Q = (x_{i_1}, x_{i_2}, \cdots, x_{i_n}) \Rightarrow X = X'Q^{-1}$$

In [ ]:
```python
import numpy as np
# A = list<list<float>> , B = list<float>
def Gauss(A,B):
    A,B = np.array(A),np.array(B)[:,np.newaxis] # to col vector
    dim = len(B)

    # return the row and col of maxValue
    def locateMax(A,depth):
        i0,j0 = np.unravel_index( np.argmax( A[depth:,depth:] ), A.shape
        i = i0 + depth
        j = j0 + depth
        return i, j

    # (i,j)<->(k,l)
    def getPQ(i, j, k, l):
        Pj = np.eye(dim)
        Qj = np.eye(dim)
        Pj[[i,k]] = Pj[[k,i]]
        Qj[:,[j,l]] = Qj[:,[l,j]]
        return Pj, Qj

    # row j += row i * k
    def MulkAndAdd(i,k,j):
        for s in range(dim):
            A[j,s] += A[i,s] * k

    Q = np.eye(dim)
    for depth in range(dim-1):
        # exchange
        max_row, max_col = locateMax(A, depth)
        Pi, Qi = getPQ(depth, depth, max_row, max_col)
        Q  = np.dot(Q, Qi)
        A = np.dot(Pi, np.dot(A, Qi))
        B = np.dot(Pi, B)
        pivot = A[depth,depth]
```

```
        for row in range(depth + 1, dim):
            p = - A[row, depth] / pivot
            MulkAndAdd(depth, p, row)
            B[row,0] += p * B[depth, 0]

    for depth in range(dim - 1, 0, -1):
        pivot = A[depth, depth]
        for row in range(depth-1, -1, -1):
            p = - A[row,depth] / pivot
            MulkAndAdd(depth, p, row)
            B[row,0] += p * B[depth, 0]

    # inverse to X
    diagA = np.diag(A)
    X = np.array([ B[i,0] / diagA[i] for i in range(dim)])
    X = np.round(X,decimals=4)
    X = np.dot(X, np.linalg.inv(Q))
    return X
```

In [ ]:
```
A = [[0.001, 2, 3],[-1, 3.712, 4.623],[-2, 1.072, 5.643]]
B = [1, 2, 3]
print("AX=B\'的解为X={}".format(Gauss(A,B)))
```

```
AX=B'的解为X=[-0.4904 -0.051   0.3675]
```

In [ ]:
```
A = [[4,2,-2],[2,2,-3],[-2,-3,14]]
B = [10,5,4]
print("AX=B\'的解为X={}".format(Gauss(A,B)))
```

```
AX=B'的解为X=[2. 2. 1.]
```

In [ ]:
```

```