Input: 代求矩阵**A**n×n ,置换矩阵**P**n×1={1, 2, 3……, n};

Output: **A**的逆矩阵**X**n×n ;

# 初始化

**U**n×n = **A, L**n×n = **E, E1 = E, E2 = E;**

# 最外层循环, 每次迭代计算**U**的一行和**L**的一列;

For i 🡨 0 ~ n-1:

*u m,i* = *u i,i* # 与最大主元行交换

For j 🡨 i ~ n-1:

*u m,i* 🡨

End For

If i != m:

Change Index(*u i,\*, u m,\**) # 交换两行索引: *u i,\** 🡨 *u m,\* , u m,\** 🡨 *u i,\* ,* **L**同理

Change Index(*l i,\*, l m,\**)

Change Item(**P**[i]*,* **P**[m])

# 交换置换矩阵**P**相应位置的元素**P***n*×1={…, *i*, …, *m*,…} 🡨 **P***n*×1={…, *m*, …, *i,* …};

End if

For j 🡨 i+1 ~ n-1: # 按行迭代

*l j,i* 🡨 *u j,i* / *u i,i* # 通过初等行变换使*uj,i*为0，同时将行变换过程中的系数保留，存放在**L**中;

For k 🡨 i ~ n-1: # 按列迭代

*u j,k* 🡨 *u j,k* *-* *u i,k* × *l j,i*

End For

End For

End For

**#** 求**U**的逆, 存放在**E1**中

For i 🡨 n-1 ~ 0: # 从最后一个对角主元反向迭代

For k 🡨 i ~ n-1: # **U**对角归一化

**E1**( *i*, *k* ) 🡨 **E1**( *i*, *k* ) / *u i*, *i*

End For

For j 🡨 i-1 ~ 0: # 按行反向迭代

For k 🡨 i ~ n-1: # 按列正向迭代

**E1**( *j*, *k* ) 🡨 **E1**( *j*, *k* ) – **E1**( *i*, *k* ) × *u*( *j*, *i* )

End For

End For

End For

**#** 求**L**的逆, 存放在**E2**中

For i 🡨 0 ~ n-1: # 从第一个对角主元正向迭代

For j 🡨 i+1 ~ n-1: # 按行正向迭代

For k 🡨 0 ~ i: # 按列正向迭代

**E2**( *j*, *k* ) 🡨 **E2**( *j*, *k* ) – **E2**( *i*, *k* ) × *l*( *j*, *i* )

End For

End For

End For

# 求**A**的逆**X**

**X** 🡨

# 按**P**对**X**的列重新排序, 注意：**P**按从后往前的顺序

For i 🡨 0 ~ n-1:

Change Index(**X**[\*, i]*,* **X**[\*, **P**[-i]])

End For

Return X # 得到结果X