

In [5]:

```
import numpy as np
def rref(A, tol=1.0e-12):
    m, n = A.shape
    i, j = 0, 0
    jb = []

    while i < m and j < n:
        # Find value and index of largest element in the remainder of column j
        k = np.argmax(np.abs(A[i:m, j])) + i
        p = np.abs(A[k, j])
        if p <= tol:
            # The column is negligible, zero it out
            A[i:m, j] = 0.0
            j += 1
        else:
            # Remember the column index
            jb.append(j)
            if i != k:
                # Swap the i-th and k-th rows
                A[[i, k], j:n] = A[[k, i], j:n]
            # Divide the pivot row i by the pivot element A[i, j]
            A[i, j:n] = A[i, j:n] / A[i, j]
            # Subtract multiples of the pivot row from all the other rows
            for k in range(m):
                if k != i:
                    A[k, j:n] -= A[k, j] * A[i, j:n]
            i += 1
            j += 1
    # Finished
    return A, jb
```

```
A = np.array([[16.0, 2, 3, 17], [5, 2, 10, 8],
              [9, 7, 6, 12], [4, 14, 15, 3]])
Areduced, jb = rref(A)
```

```
print(f"The matrix as rank {len(jb)}")  
print(Areduced)
```

```
The matrix as rank 4  
[[1. 0. 0. 0.]  
 [0. 1. 0. 0.]  
 [0. 0. 1. 0.]  
 [0. 0. 0. 1.]]
```

In []: