矩阵乘法的基本原理是：对于两个矩阵A和B，其乘积C是一个新的矩阵，其行数是A的行数，列数是B的列数。C的每个元素是A的对应行和B的对应列的乘积之和。

**自然语言描述**

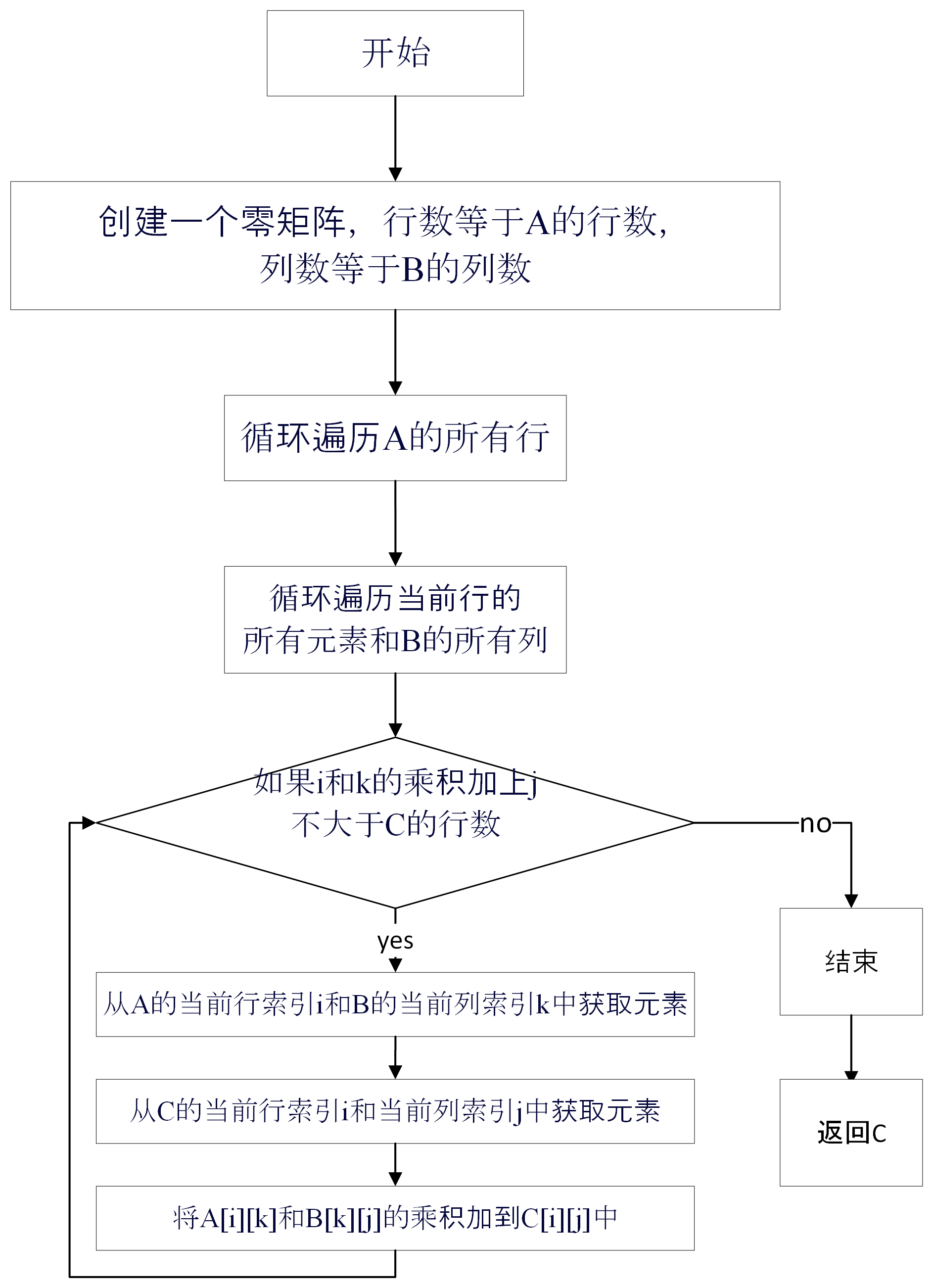
初始化一个零矩阵，其行数等于A的行数，列数等于B的列数。

遍历A的行，假设当前行索引为i，遍历该行元素和B的所有列，假设当前列索引为j。

对于每个元素A[i][k]和B[k][j]，计算乘积并加到新矩阵的C[i][j]中。

完成所有乘积和的计算后，返回新矩阵C。

**流程图描述**



**伪代码描述**

初始化一个零矩阵C，行数为A的行数，列数为B的列数

对于i从0到(A的行数-1):

对于j从0到(B的列数-1):

对于k从0到(A的列数-1):

C[i][j] += A[i][k] \* B[k][j]

返回C

**时间和空间复杂度分析**

时间复杂度：O(n3)，其中n是矩阵A和B的乘积。这是因为需要遍历A的所有行，然后遍历每行的所有元素和B的所有列，然后进行乘法和加法运算。

空间复杂度：O(n2)，这是因为在计算过程中创建了一个新的矩阵C来存储结果，其大小取决于输入矩阵A和B的大小。