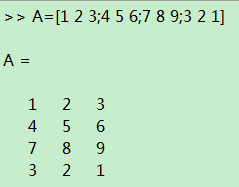
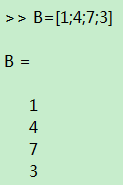
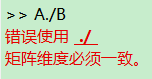
**bsxfun（fun,A,B）**

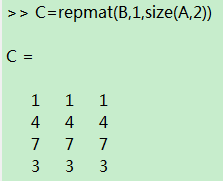
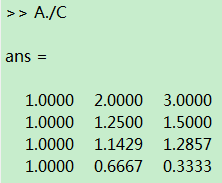
偶然间发现了这个函数，强大得不得了呀，它的作用是：对两个矩阵A和B之间的每一个元素进行指定的计算（函数fun指定）；并且具有自动扩维的作用

例如，A是一个4\*3的矩阵，B是一个4\*1的列向量，如果想要矩阵A的每一列都对向量B进行一些操作，比如，A的每一列的相应元素都要除以向量B中的相应元素，那么，比较原始的方法就是利用repmat命令，将向量B进行平铺，对它的每一列进行复制，生成一个新的矩阵C，该矩阵C的每一列都是向量B，然后，利用点除，令A点除B，就实现了矩阵A的每一列的元素都除以向量B的元素的要求；

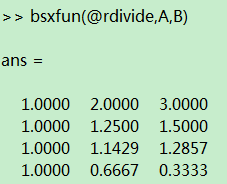
需要A的每一列都除以向量B，如果直接用点除，则会出错，因为点乘要求两个操作对象维度一致；

这是，比较原始的方法是：

但是，用这种方法会存在一个问题：如果数据的尾数过大，该方法会出现内存上的问题；

那么，bsxfun函数就是解决这个问题的great method=====bsxfun(@rdivide,A,B)



该函数的具体实现过程是：

①判断A和B的维度是否相同，如果相同，直接A./B；

@如果A和B的维度不同，则A或者B必须有一个在某个维度上是1，比如，上例中的B在列方向维度是1，那么，该函数将会对列向量B在列方向上进行复制，使其与矩阵A具有相同的列维度，然后，再进行A./B维度扩充后的矩阵

\*\*\*\*\*matlab带有的函数操作如下\*\*\*\*\*

