**题目描述**

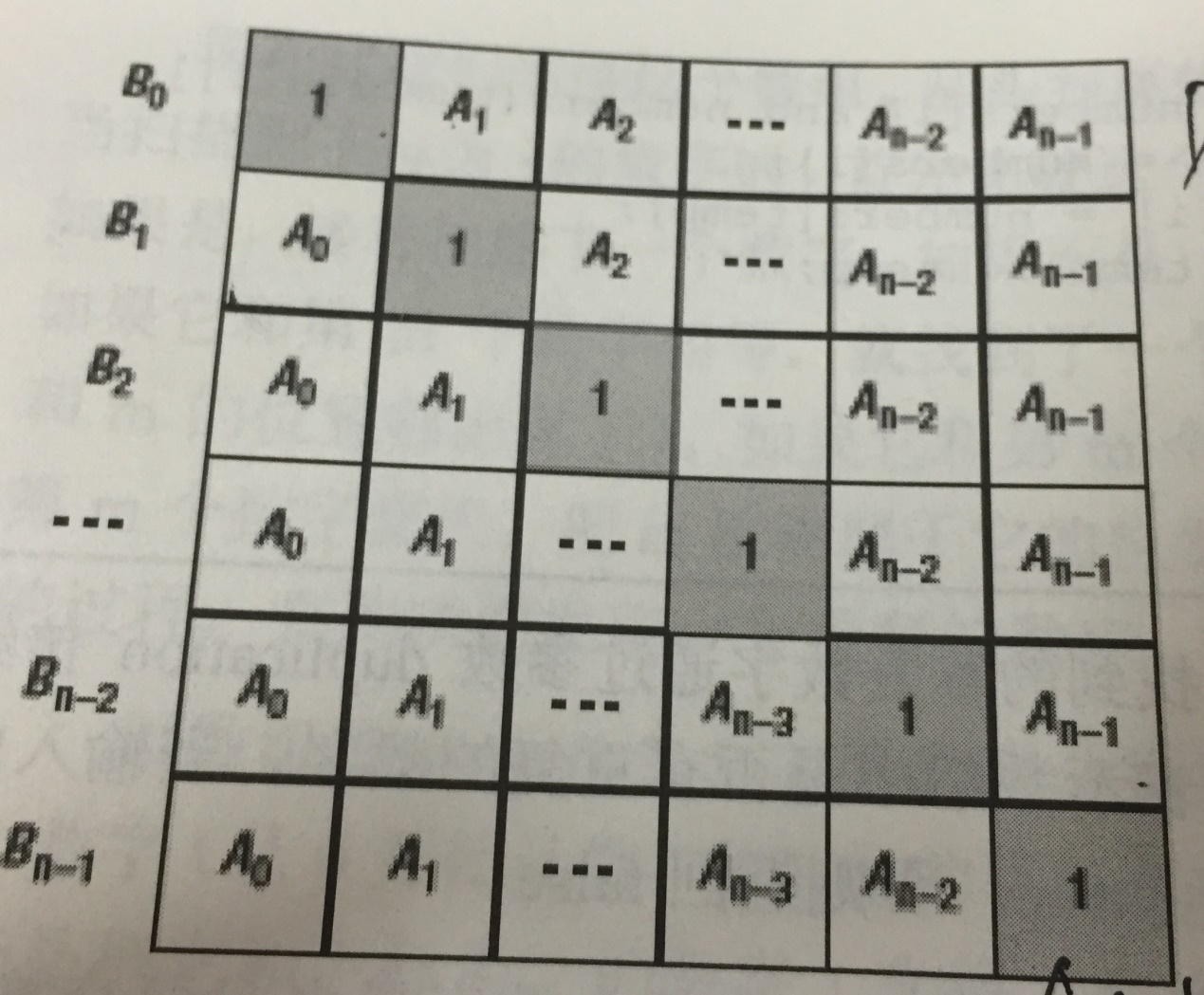
给定一个数组A[0,1,...,n-1],请构建一个数组B[0,1,...,n-1],其中B中的元素B[i]=A[0]\*A[1]\*...\*A[i-1]\*A[i+1]\*...\*A[n-1]。不能使用除法。

剑指的思路：

**B[i]的值可以看作下图的矩阵中每行的乘积。**

下三角用连乘可以很容求得，上三角，从下向上也是连乘。

因此我们的思路就很清晰了，先算下三角中的连乘，即我们先算出B[i]中的一部分，然后倒过来按上三角中的分布规律，把另一部分也乘进去。



import java.util.ArrayList;

public class Solution {

public int[] multiply(int[] A) {

int n = A.length;

int[] B = new int[n];

B[0]=1;

for (int i=1;i<n;i++){

B[i]=B[i-1]\*A[i-1];

}

int temp = 1;

for(int j=n-2;j>=0;j--){

temp = temp \* A[j+1];

B[j] = B[j]\*temp;

}

return B;

}

}