### 伪代码描述：

1

//给定p=(p0,p1,...pn),其中pi-1和pi分别为矩阵Ai的行数和列数  
//s[i][j]中存储了Ai...Aj的最小乘法次数对应的分割点k  
//m[i][j]中存储了Ai...Aj的最小乘法次数  
//length表示矩阵链的长度

MATRIX-CHAIN-ORDER(p,length,m,s)

n = length – 1

for i = 1 to n do

m[i, i] = 0

for l = 2 to n do // l为子矩阵链的长度，依次考虑由小到大的子矩阵链

for i = 1 to n-l+1 do

j = i+l-1

m[i, j] = ∞

for k = i to j–1 do

q = m[i, k] + m[k+1, j] + pi-1pkpj

if q < m[i, j] then

m[i, j] = q

s[i, j] = k

2

//给定前述过程MATRIX-CHAIN-ORDER输出的表s

//下列过程打印矩阵链的最优加括号方案

PRINT-OPTIMAL-PARENS(s, i, j)

if i = j then

print “A”i

else

print “(“

PRINT-OPTIMAL-PARENS(s, i, s[i, j])

PRINT-OPTIMAL-PARENS(s, s[i, j]+1, j)

print “)

### 具体输入格式：

需要修改p数组中的值

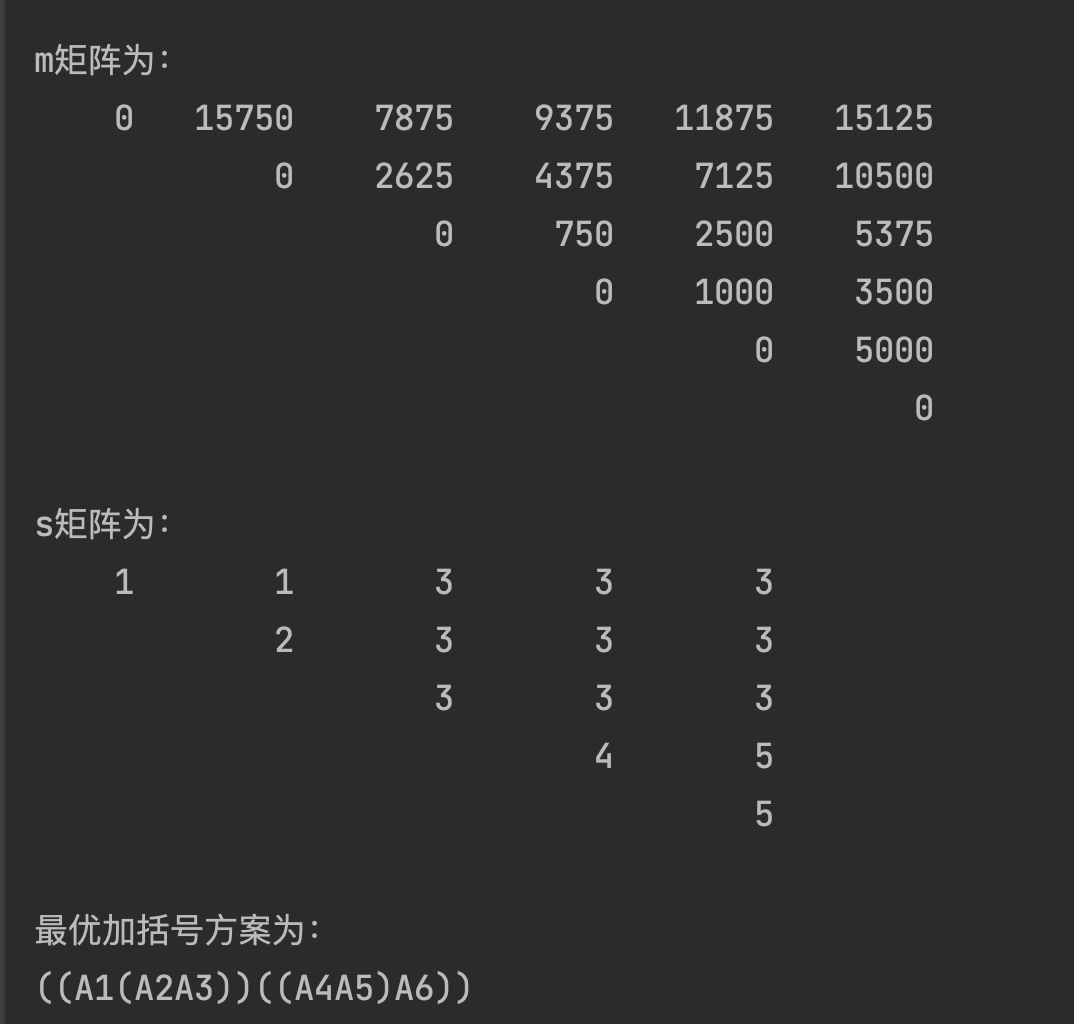


修改常量M为p中元素个数



### 具体输出格式：

分别输出m和s矩阵以及最优加括号方案。



### 测试用例：

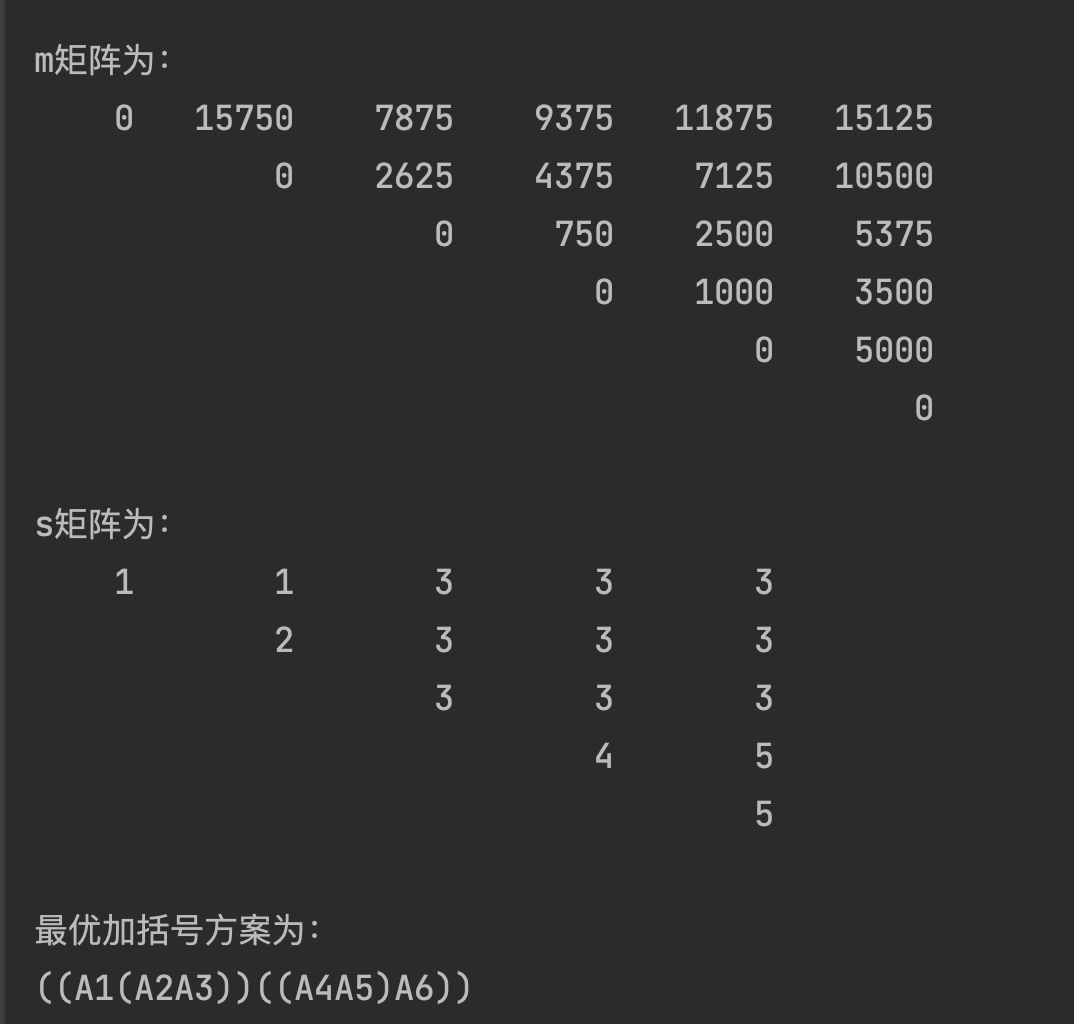
1.

输入：





输出：



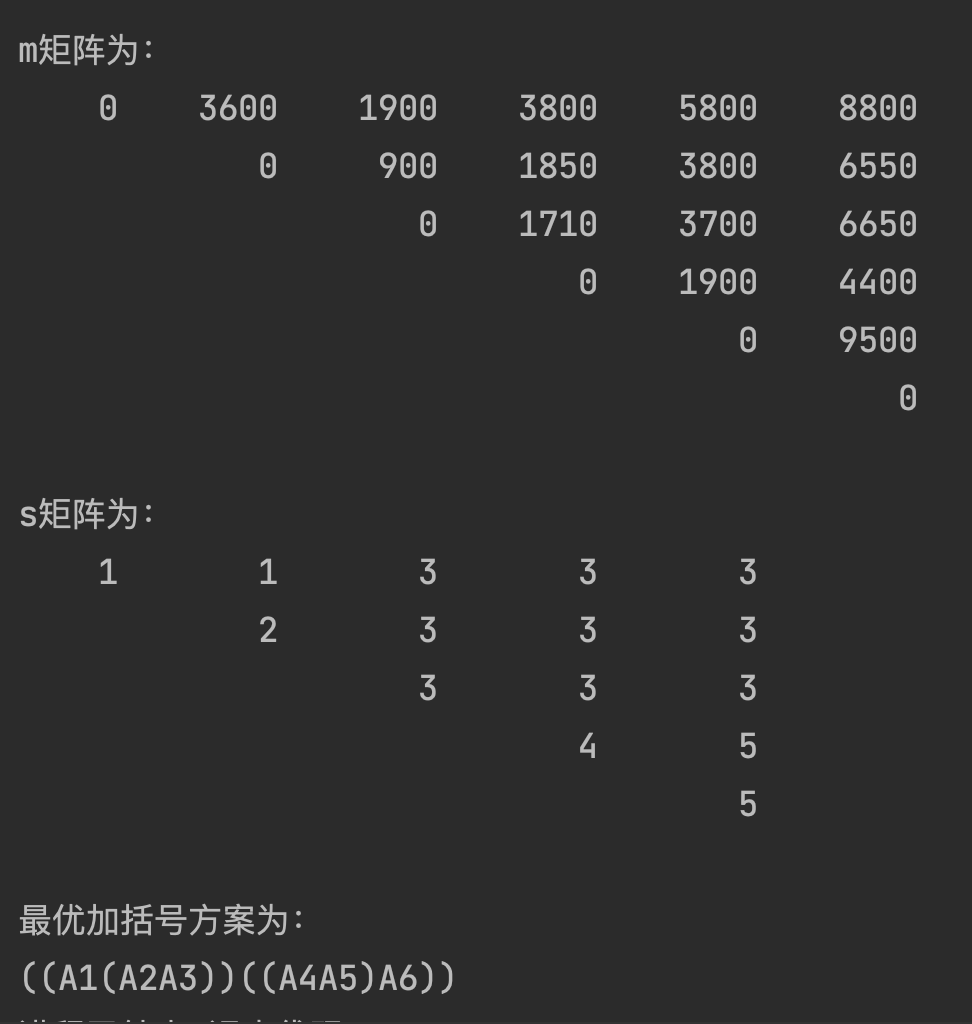
2

输入：





输出：



3

输入：





输出：

