20174179 杨小川 计科卓越班

15.2-1 对矩阵规模序列<5,10,3,12,5,50,6>，求矩阵链最优括号化方案。

**#include** <iostream>

**#include** <string>

**#include** <iomanip>

**#define** UNC 99999

**using** **namespace** std;

**static** string matirx\_A[7] = {"A0","A1","A2","A3","A4","A5","A6"};

**static** **int** matrix\_size[7] = {5,10,3,12,5,50,6};

**int**\*\* **MATRIX\_CHAIN\_ORDER**(**int** \*\*m, **int** \*\*s, **int** \*p, **int** n)

{

**int** i,j,k,l;

**for**(i = 1; i <= n; i++)

m[i][i] = 0;

**for**(l = 2; l <= n; l++){

**for**(i = 1; i <= n-l+1; i++){

j = i+l-1;

m[i][j] = UNC;

**for**(k = i; k <= j-1; k++) {

**int** q = m[i][k] + m[k+1][j] + p[i-1] \* p[k] \* p[j];

**if**(q < m[i][j]){

m[i][j] = q;

s[i][j] = k;

}

}

}

}

// Print the matrix

**for**(i = 1; i<= n;i++)

{

**for** (j= 1;j<=n;j++)

**if** (m[i][j] != UNC)

{

cout << right << setw(8) << m[i][j];

}

**else**

{

cout << right << setw(8) << " ";

}

cout<<**endl**;

}

**return** s;

}

// Print the optimal parenthesis scheme

**void** **PRINT\_PSRENTHSIS**(**int** \*\*s, **int** i,**int** j)

{

**if**(i == j)

cout << matirx\_A[i];

**else**

{

cout << "(";

PRINT\_PSRENTHSIS(s, i, s[i][j]);

PRINT\_PSRENTHSIS(s, s[i][j] + 1, j);

cout << ")";

}

}

**int**\*\* **INIT\_MATRIX**(**int** row,**int** col)

{

**int** \*\*matrix;

matrix = (**int**\*\*)**malloc**(**sizeof**(**int**\*)\*row);

**for**(**int** i = 0 ; i < row; i++)

matrix[i] = (**int**\*)**malloc**(**sizeof**(**int**)\*col);

**for**(**int** i = 0;i<row;i++)

**for**(**int** j =0;j<col;j++)

matrix[i][j] = UNC;

**return** matrix;

}

**int** **main**() {

**int** \*\*s;

**int** \*\*m;

**int** n = **sizeof**(matrix\_size)/**sizeof**(**int**);

m = INIT\_MATRIX(n,n);

s = INIT\_MATRIX(n,n);

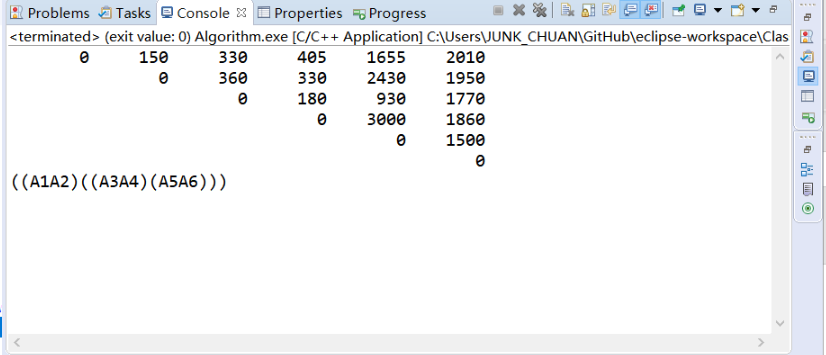
s = MATRIX\_CHAIN\_ORDER(m, s, matrix\_size, n-1);

PRINT\_PSRENTHSIS(s,1,6);

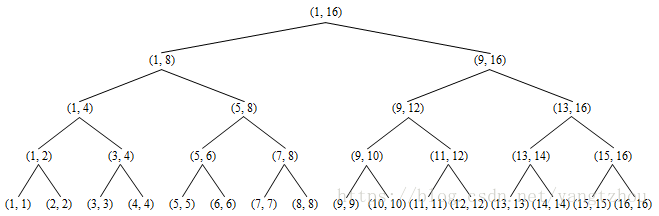
**return** 0;

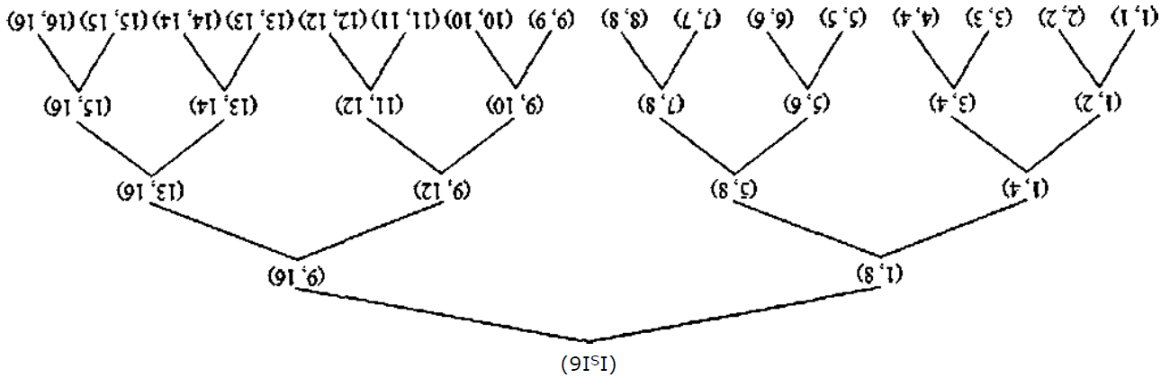
}

输出结果：



15.3-2对一个16个元素的数组，画出2.3.1节中MERGE-SORT过程运行的递归调用树。解释备忘技术为什么对MERGE-SORT这种分治算法无效。





MERGE-SORT虽也是把问题划分为子问题，然而子问题之间互相不重叠，因此不需要采用备忘法。

15.3-3考虑矩阵链乘法问题的一个变形：目标改为最大化矩阵序列括号化方案的标量乘法运算次数，而非最小化。此问题具有最优子结构性质吗？

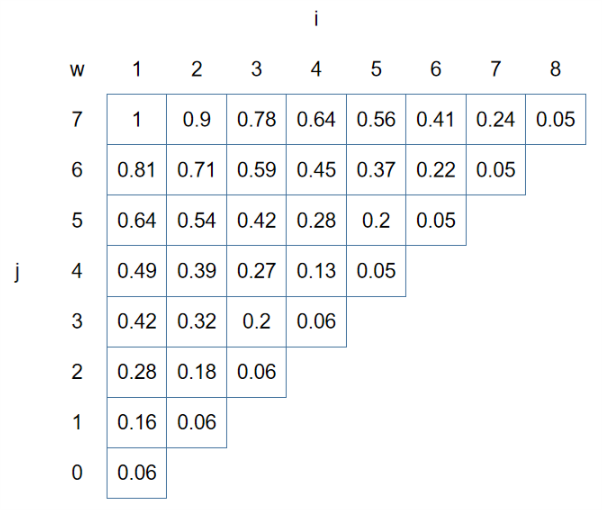
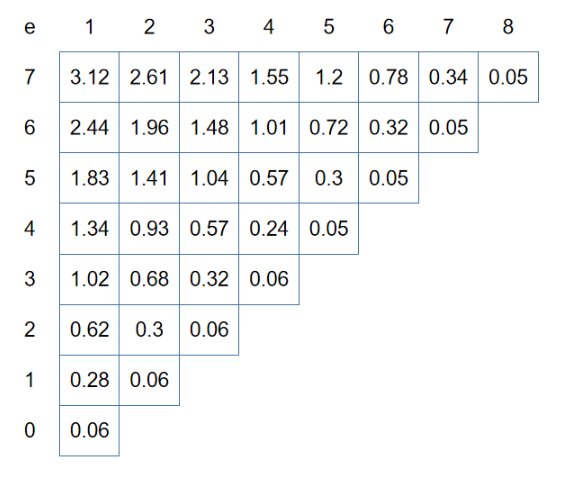
 此问题也具有最优子结构性质。用m[i, j]表示计算矩阵链乘法所需要的最大计算代价。那么代价最大的完全括号化方案的子问题为m[i, j] = m[i, k] + m[k+1, j] + pi-1pkpj。下面给出递归式。



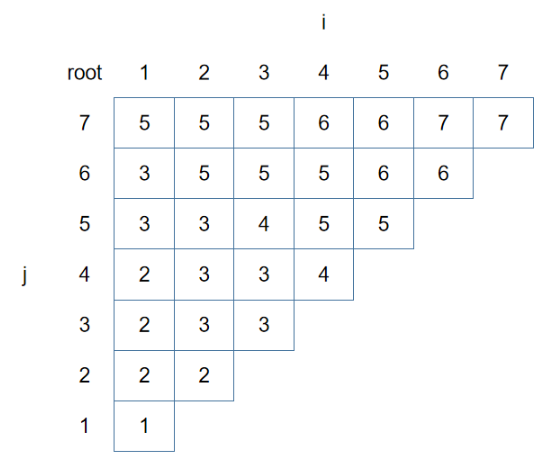
15.5-2若7 77个关键字的概率如下所示，求其最优二叉搜索树的结构和代价。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| i | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| pi |  | 0.04 | 0.06 | 0.08 | 0.02 | 0.10 | 0.12 | 0.14 |
| qi | 0.06 | 0.06 | 0.06 | 0.06 | 0.05 | 0.05 | 0.05 | 0.05 |

最优二叉搜索树如下所示。期望搜索代价为3.12。



最佳根节点划分为：



最优的BST结构为：

