**西安电子科技大学**

**算法设计与分析 课程实验报告**

计算机科学与技术 学院 1903052 班

**实验名称**

姓名 赵书晨 学号 19030500122

实验日期 2022 年 4 月 6 日

### 一、实验内容

##### （一）Matrix-chain product

1.最优括号化方案的结构特征

用记号A[i..j]表示乘积A[i]A[i+1]..A[j]求值的结果，其中i <=j 。

假设A[i]A[i+1]...A[j]的一个最优解括号把乘积在A[k]和A[k+1]之间分开，则对A[i]A[i+1]...A[j]最优解括号化方案中的“前缀”子链A[i]A[i+1]...A[k]的最优括号化的方法，必须是A[i]A[i+1]...A[k]的一个最有解括号化方案，类似的，A[k+1]A[k+2]…A[j]同理。

2.设m[i][j]为计算矩阵A[i..j]所需的标量乘法运算次数的最小值；

对整个问题，计算A[1..n]的最小代价就是m[1][n]。

假设最优加全部括号将乘积A[i]A[i+1]...A[j]从A[k]和A[k+1]之间分开，i <= k < j。

则：m[i][j] = m[i][k] + m[k+1][j] + p[i-1]\*p[k]\*p[j]

关于对乘积A[i]A[i+1]...A[j]加全部括号的最小代价的递归定义为:

m[i][j] = 0 if i == j

m[i][j] = min(i<=k<j){m[i][k] + m[k+1][j] + p[i-1]\*p[k]\*p[j]} s[i][j]=k if i < j

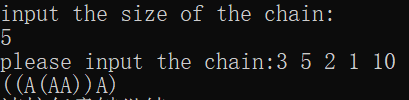
用s[i][j]记录最优值m[i][j]的对应的分割点。

3.用迭代自底向上的表格法来计算最优代价。

4.利用保存在表格s[n][n]内的、经过计算的信息来构造一个最优解。

按最优方式计算A[1..n]时，最终矩阵相乘次序是A[1..s[1][n]]A[a[1][n]+1..n]。

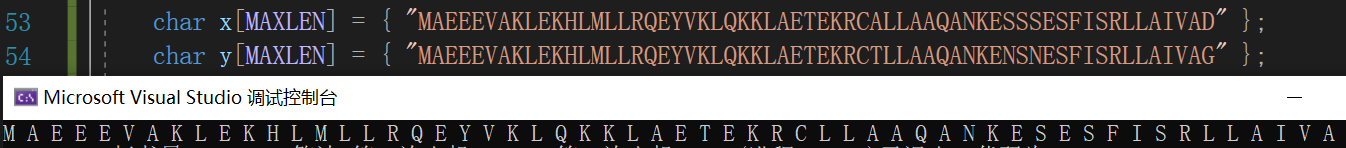
之前的乘法可以递归地进行。



##### （二）Longest Common Subsequence

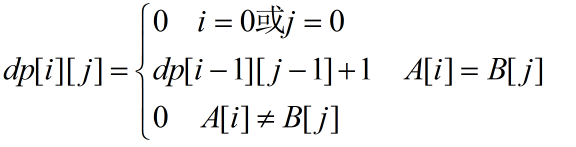
最长公共子串，不要求最长公共子串必须连续出现在两个字符串中，只需要能保持顺序的出现在序列中即可。

IMG_256



##### （三）Longest Common Substring

定义dp[i][j]的含义是：字符串 [a1,a2,...,ai]与字符串[b1,b2,...,bj]的最长公共连续子串的最后一个字符与这个两个字符串的最后一个字符相等的情况下，这个LCS的长度。因此状态转移方程为：



A[i]!=B[j]的情况下，dp[i][j]也等于0，这是子序列和子串在状态转移方程上的区别。

那么，我们通过两层循环，计算出dp[][]的值，然后找到其中最大的值，就是LCS的长度了。

##### （四）Max Sum

1.首先，读取第一个数据，令now和max等于第一个数据，初始化pos1,pos2,x位置

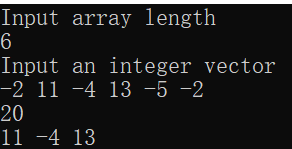
2.然后，读入第二个数据，判断

①若是now+temp<temp，表示当前读入的数据比之前存储的加上当前的还大，说明可以在当前另外开始记录，更新now=temp

②反之，则表示之前的数据和在增大，更新now=now+temp

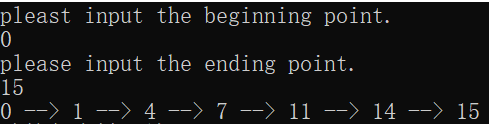
3.之后，把now跟max做比较，更新或者不更新max的值，记录起始、末了位置。

4.循环2~3步骤，直至读取数据完毕。



##### （五）Shortest path in multistage graphs

使用djstra算法。



### 三、实验中遇到的问题

动态规划问题的主要难点在于，发现重叠子问题，以及状态的定义和状态转移。

### 四、实验小结

通过本次实验，了解并掌握了动态规划思想：即自底向上，站在已知的角度，通过定位已知和未知之间的关系，一步一步向前推导，进而求解出未知的值。