

题目 1:

X 和 Y 是由 26 个大写字母表中的字母组成的序列，长度分别为 L_x 和 L_y 。为了比较它们的相似性，我们需要按照下列公式计算出这两条序列的相似度矩阵 M ：

$$M(i, j) = \max \begin{cases} M(i-1, j-1) + S(X_i, Y_j), \\ M(k, j) + d, k = 0, \dots, i-1, \\ M(i, k) + d, k = 0, \dots, j-1 \end{cases}$$

其中 M 的维度是 $L_x \times L_y$ ， $1 \leq i \leq L_x$ ， $1 \leq j \leq L_y$ ， d 是一个常数， S 是一个计算字母之间关系的常数矩阵，其维度是 26×26 ， M 的第 0 行和第 0 列按照下列方法初始化：

$$M(0, p) = p \times d, 0 \leq p \leq L_x$$

$$M(q, 0) = q \times d, 0 \leq q \leq L_y$$

现在给出下列条件：

常数 d 的值是 -8，常数矩阵 S 给定如下

```
int S[26][26] =
/* A B C D E F G H I J K L M N O P Q R S T U V W X Y Z */
{ /* A */ { 4, -2, 0, -2, -1, -2, 0, -2, -1, 0, -1, -1, -1, -2, 0, -1, -1, -1, 1, 0, 0, 0, -3, 0, -2, -1},
  /* B */ { -2, 4, -3, 4, 1, -3, -1, 0, -3, 0, 0, -4, -3, 3, 0, -2, 0, -1, 0, -1, 0, -3, -4, -1, -3, 1},
  /* C */ { 0, -3, 9, -3, -4, -2, -3, -3, -1, 0, -3, -1, -1, -3, 0, -3, -3, -3, -1, -1, 0, -1, -2, -2, -2, -3},
  /* D */ { -2, 4, -3, 6, 2, -3, -1, -1, -3, 0, -1, -4, -3, 1, 0, -1, 0, -2, 0, -1, 0, -3, -4, -1, -3, 1},
  /* E */ { -1, 1, -4, 2, 5, -3, -2, 0, -3, 0, 1, -3, -2, 0, 0, -1, 2, 0, 0, -1, 0, -2, -3, -1, -2, 4},
  /* F */ { -2, -3, -2, -3, -3, 6, -3, -1, 0, 0, -3, 0, 0, -3, 0, -4, -3, -3, -2, -2, 0, -1, 1, -1, 3, -3},
  /* G */ { 0, -1, -3, -1, -2, -3, 6, -2, -4, 0, -2, -4, -3, 0, 0, -2, -2, -2, 0, -2, 0, -3, -2, -1, -3, -2},
  /* H */ { -2, 0, -3, -1, 0, -1, -2, 8, -3, 0, -1, -3, -2, 1, 0, -2, 0, 0, -1, -2, 0, -3, -2, -1, 2, 0},
  /* I */ { -1, -3, -1, -3, -3, 0, -4, -3, 4, 0, -3, 2, 1, -3, 0, -3, -3, -3, -2, -1, 0, 3, -3, -1, -1, -3},
  /* J */ { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  /* K */ { -1, 0, -3, -1, 1, -3, -2, -1, -3, 0, 5, -2, -1, 0, 0, -1, 1, 2, 0, -1, 0, -2, -3, -1, -2, 1},
  /* L */ { -1, -4, -1, -4, -3, 0, -4, -3, 2, 0, -2, 4, 2, -3, 0, -3, -2, -2, -2, -1, 0, 1, -2, -1, -1, -3},
  /* M */ { -1, -3, -1, -3, -2, 0, -3, -2, 1, 0, -1, 2, 5, -2, 0, -2, 0, -1, -1, -1, 0, 1, -1, -1, -1, -1},
  /* N */ { -2, 3, -3, 1, 0, -3, 0, 1, -3, 0, 0, -3, -2, 6, 0, -2, 0, 0, 1, 0, 0, -3, -4, -1, -2, 0},
  /* O */ { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  /* P */ { -1, -2, -3, -1, -1, -4, -2, -2, -3, 0, -1, -3, -2, -2, 0, 7, -1, -2, -1, -1, 0, -2, -4, -2, -3, -1},
  /* Q */ { -1, 0, -3, 0, 2, -3, -2, 0, -3, 0, 1, -2, 0, 0, 0, -1, 5, 1, 0, -1, 0, -2, -2, -1, -1, 3},
  /* R */ { -1, -1, -3, -2, 0, -3, -2, 0, -3, 0, 2, -2, -1, 0, 0, -2, 1, 5, -1, -1, 0, -3, -3, -1, -2, 0},
  /* S */ { 1, 0, -1, 0, 0, -2, 0, -1, -2, 0, 0, -2, -1, 1, 0, -1, 0, -1, 4, 1, 0, -2, -3, 0, -2, 0},
  /* T */ { 0, -1, -1, -1, -1, -2, -2, -2, -1, 0, -1, -1, -1, 0, 0, -1, -1, -1, 1, 5, 0, 0, -2, 0, -2, -1},
  /* U */ { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  /* V */ { 0, -3, -1, -3, -2, -1, -3, -3, 3, 0, -2, 1, 1, -3, 0, -2, -2, -3, -2, 0, 0, 4, -3, -1, -1, -2},
  /* W */ { -3, -4, -2, -4, -3, 1, -2, -2, -3, 0, -3, -2, -1, -4, 0, -4, -2, -3, -3, -2, 0, -3, 11, -2, 2, -3},
  /* X */ { 0, -1, -2, -1, -1, -1, -1, -1, -1, 0, -1, -1, -1, -1, 0, -2, -1, -1, 0, 0, 0, -1, -2, -1, -1, -1},
  /* Y */ { -2, -3, -2, -3, -2, 3, -3, 2, -1, 0, -2, -1, -1, -2, 0, -3, -1, -2, -2, -2, 0, -1, 2, -1, 7, -2},
  /* Z */ { -1, 1, -3, 1, 4, -3, -2, 0, -3, 0, 1, -3, -1, 0, 0, -1, 3, 0, 0, -1, 0, -2, -3, -1, -2, 4}
};
```

序列 X 和 Y 具有相同的长度（该长度记为变量 dimension）并且要求使用下面的伪代码生成这两个序列：

```
char residues[27] = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
```

```
X = (char*)malloc(sizeof(char)*(dimension));
```

```

Y = (char*)malloc(sizeof(char)*(dimension));
for (i = 0; i < dimension; i++)
{
    X[i] = residues[i%26];
    Y[i] = residues[i%26];
}

```

根据上面给出的条件，当 dimension 的值为 7 时可以计算出如下的相似度矩阵 M :

	A	B	C	D	E	F	G
0	-8	-16	-24	-32	-40	-48	-56
A	-8	4	-4	-4	-4	-4	-4
B	-16	-4	8	0	0	0	0
C	-24	-4	0	17	9	9	9
D	-32	-4	0	9	23	15	15
E	-40	-4	0	9	15	28	20
F	-48	-4	0	9	15	20	34
G	-56	-4	0	9	15	20	26

要求:

计算当 dimension 的值分别为 3000 和 5000 时的相似度矩阵 M ，要求:

1. 必须计算全部矩阵：最终测试的时候我们会改变序列 X 和 Y 的生成方式使得两条序列不相同，此时 M 不沿对角线对称，需要计算全部矩阵单元的值。
2. 计算结果正确
为了验证计算结果，所编写的程序要求至少可以处理以下两个参数：dimension 和 M 的行号 row_number
例如假设编译好的最终程序是 a.out，那么命令行：./a.out 2000 200 表示计算当 dimension 为 2000 时的相似度矩阵，并且把最终计算出的矩阵的第 200 行打印出来以检查结果是否正确。
3. 使用多核多线程或者其它并程序序设计方法使得当程序运行在具有相同计算能力的 CPU 上时，计算时间尽可能少。

题目 2:

要求在完成题目 1 的基础上，按照下列新给出的公式再计算一遍当 dimension 为 3000 和 5000 时的相似度矩阵 M :

$$M(i, j) = \max \begin{cases} M(i-1, j-1) + S(X_i, Y_j), \\ M(k, j) + (i-k)^2, k = 0, \dots, i-1, \\ M(i, k) + (j-k)^2, k = 0, \dots, j-1 \end{cases}$$

所有其它在题目 1 中给定的已知条件和参数不做变化，因此按照上述新的公式和其它给定条件当 dimension 的值为 7 时可以计算出如下的相似度矩阵 M :

	A	B	C	D	E	F	G
0	-8	-16	-24	-32	-40	-48	-56
A	-8	4	5	8	13	20	29
B	-16	5	8	9	14	21	30
C	-24	8	9	17	18	24	33
D	-32	13	14	18	23	29	38
E	-40	20	21	24	29	36	45
F	-48	29	30	33	38	45	54
G	-56	41	42	45	50	57	66