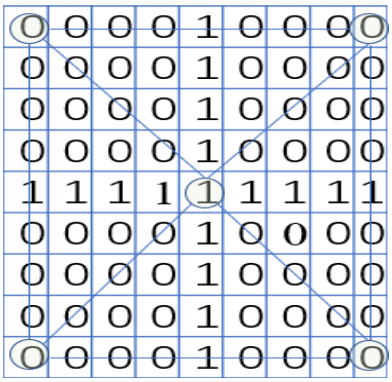
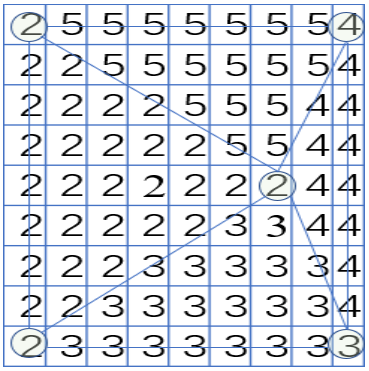
计算机视觉与模式识别作业六（2021 版本）

姓名：	李智骋	学号：	2196113526
班级：	计算机95	得分：	

1、 给定源图像和目标图像如下所示



源图像



目标图像

① 计算特征点在 t=0.25,0.5 和 0.75 时候的平均形状；

根据公式，给定原图像和目标图像的特征点对(p_1, p_2)，t时候的位置

$$p_t = (1-t) * p_1 + t * p_2$$

假设图像左上角为原点，水平方向为x轴，竖直方向为y轴，图像中的点从1开始计数，那么五个特征点的坐标以及不同t值下的坐标：

原图像特征点	(1,1)	(1,9)	(5,5)	(9,1)	(9,9)
$t = 0.25$	(1,1)	(1,9)	(5.5,5)	(9,1)	(9,9)
$t = 0.5$	(1,1)	(1,9)	(6,5)	(9,1)	(9,9)
$t = 0.75$	(1,1)	(1,9)	(6.5,5)	(9,1)	(9,9)
目标图像特征点	(1,1)	(1,9)	(7,5)	(9,1)	(9,9)

② 计算 t=0.5 时，源图像和目标图像向平均形状的变形结果；（图像变形均采用最近邻插值）

根据变形后的点(x', y')，存在点所对应的三角面片（由A、B、C表示），相应的重心坐标满足：

$$\begin{bmatrix} A_x^{'} & B_x^{'} & C_x^{'} \\ A_y^{'} & B_y^{'} & C_y^{'} \\ 1 & 1 & 1 \end{bmatrix} * \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} x^{'} \\ y^{'} \\ 1 \end{bmatrix}$$

根据重心坐标在仿射变换下不变的原理，计算变形前的点(x, y), 采用最邻近插值：

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = Round(\begin{bmatrix} A_x & B_x & C_x \\ A_y & B_y & C_y \\ 1 & 1 & 1 \end{bmatrix} * \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix})$$

源图像向平均形状的变形结果：

0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0	0
1	1	1	1	1	1	1	1	1
0	0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0

目标图像向平均形状的变形结果：

2	5	5	5	5	5	5	5	4
2	2	5	5	5	5	5	5	4
2	2	2	5	5	5	4	4	4
2	2	2	2	5	5	4	4	4
2	2	2	2	2	2	4	4	4
2	2	2	2	3	3	4	4	4
2	2	2	3	3	3	3	4	4
2	2	3	3	3	3	3	3	4
2	3	3	3	3	3	3	3	3

③ 计算 t=0.5 时的平均图像

根据公式，给定原图像和目标图像的像素对(q_1, q_2)，t平均时候的像素q_t

$$q_t = (1 - t) * q_1 + t * q_2$$

此时的平均图像：

1.0000	2.5000	2.5000	2.5000	3.0000	2.5000	2.5000	2.5000	2.0000
1.0000	1.0000	2.5000	2.5000	3.0000	2.5000	2.5000	2.5000	2.0000
1.0000	1.0000	1.0000	2.5000	3.0000	2.5000	2.0000	2.0000	2.0000
1.0000	1.0000	1.0000	1.0000	2.5000	3.0000	2.0000	2.0000	2.0000
1.5000	1.5000	1.5000	1.5000	1.5000	1.5000	2.5000	2.5000	2.5000
1.0000	1.0000	1.0000	1.0000	1.5000	2.0000	2.0000	2.0000	2.0000
1.0000	1.0000	1.0000	1.5000	2.0000	1.5000	1.5000	2.0000	2.0000
1.0000	1.0000	1.5000	1.5000	2.0000	1.5000	1.5000	1.5000	2.0000
1.0000	1.5000	1.5000	1.5000	2.0000	1.5000	1.5000	1.5000	1.5000

出于图像像素值常为整数，若对平均图像四舍五入取整，结果如下：

1	3	3	3	3	3	3	3	2
1	1	3	3	3	3	3	3	2
1	1	1	3	3	3	2	2	2
1	1	1	1	3	3	2	2	2
2	2	2	2	2	2	3	3	3
1	1	1	1	2	2	2	2	2
1	1	1	2	2	2	2	2	2
1	1	2	2	2	2	2	2	2
1	2	2	2	2	2	2	2	2

使用matlab编程如下，**在判断像素点所处的三角面片时，使用了简便的方法——通过相应像素点在目标图像的取值判断，但由于是后向变换，像素点所在面片不应取决于目标图像的特征点分布，而应取决于$t = 0.5$时的图像特征点分布，计算存在较小的误差：**

```
clc;clear;
SP = zeros(9);
for i = 1 : 1 : 9
    SP(5, i) = 1;
    SP(i, 5) = 1;
end
DP = [2, 5, 5, 5, 5, 5, 5, 5, 4;
    2, 2, 5, 5, 5, 5, 5, 5, 4;
    2, 2, 2, 2, 5, 5, 5, 4, 4;
    2, 2, 2, 2, 2, 5, 5, 4, 4;
    2, 2, 2, 2, 2, 2, 2, 4, 4;
    2, 2, 2, 2, 2, 3, 3, 4, 4;
    2, 2, 2, 3, 3, 3, 3, 3, 4;
    2, 2, 3, 3, 3, 3, 3, 3, 4;
    2, 3, 3, 3, 3, 3, 3, 3, 3];
tranSP = zeros(9);
tranDP = zeros(9);

for i = 1 : 1 : 9
    for j = 1 : 1 : 9
        if DP(j, i) == 2
            Ax = 1;Bx = 1;Ay = 1;By = 9;
        elseif DP(j, i) == 3
            Ax = 1;Bx = 9;Ay = 9;By = 9;
        elseif DP(j, i) == 4
            Ax = 9;Bx = 9;Ay = 9;By = 1;
        else
            Ax = 9;Bx = 1;Ay = 1;By = 1;
        end
        alpha = [Ax,Bx,6;Ay,By,5;1,1,1]\[i;j;1];
        PreSP = [Ax,Bx,5;Ay,By,5] * alpha;
        tranSP(j,i) = SP(round(PreSP(2)),round(PreSP(1)));
        PreDp = [Ax,Bx,7;Ay,By,5] * alpha;
        tranDP(j,i) = DP(round(PreDp(2)),round(PreDp(1)));
    end
end
disp(tranSP);
disp(tranDP);
disp(0.5*tranSP+0.5*tranDP)
```

2、 给定一个图像如下所示：

0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
127	127	127	127	255	127	127	127	127
255	255	255	255	255	255	255	255	255
127	127	127	127	255	127	127	127	127
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0

① 它在 X 和 Y 方向的梯度分别是多少？

由于并没有指定模板，我分别采用[-1, 1]和[-1, 0, 1]两种模板计算这一道题。

假设图像左上角为原点，水平方向为x轴，竖直方向为y轴，图像中的像素点从1开始计数，**梯度计算采用向前差分（即运算模板为$[-1, 1]$、$[-1, 1]^T$），向前差分时不再有下一个像素点用于计算时，默认对应梯度为0(即边界条件采用最邻近像素点填充)。**

计算所得x方向梯度：

x方向梯度：

0	0	127	128	-128	-127	0	0	0
0	0	127	128	-128	-127	0	0	0
0	0	127	128	-128	-127	0	0	0
0	0	0	128	-128	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	128	-128	0	0	0	0
0	0	127	128	-128	-127	0	0	0
0	0	127	128	-128	-127	0	0	0
0	0	127	128	-128	-127	0	0	0

计算所得y方向梯度：

y方向梯度：

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
127	127	127	0	0	0	127	127	127
128	128	128	128	0	128	128	128	128
-128	-128	-128	-128	0	-128	-128	-128	-128
-127	-127	-127	0	0	0	-127	-127	-127
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	0

② 利用 L1 范数计算能量值，请问它的能量矩阵是多少？

二维向量的L1范式计算能量：

$$e(x,y) = |dx(x,y)| + |dy(x,y)|$$

根据(1)中结果，计算的能量矩阵为：

能量矩阵：

0	0	127	128	128	127	0	0	0
0	0	127	128	128	127	0	0	0
127	127	254	128	128	127	127	127	127
128	128	128	256	128	128	128	128	128
128	128	128	128	0	128	128	128	128
127	127	127	128	128	0	127	127	127
0	0	127	128	128	127	0	0	0
0	0	127	128	128	127	0	0	0
0	0	127	128	128	127	0	0	0

③ 利用动态规划计算 x 方向的 Seam，请问初始化后的值函数矩阵和路径矩阵分别是多少；

计算x方向的Seam，即求取一个横向的路径使得能量最小，假设相邻被选择的像素点y方向的差值不大于1。

初始条件：

$$\begin{cases} V(1,y) = e(1,y) \\ P(1,y) = 0 \end{cases}, y = 1 \dots 9$$

初始化的值函数矩阵：

0
0
127
128
128
127
0
0
0

初始化的路径矩阵：

0
0
0
0
0
0
0
0
0

④ 请问最终的值函数矩阵和路径矩阵分别是多少；

更新：

$$\begin{cases} V(x,y) = e(x,y) + \min\left(e(x-1,y-1), e(x-1,y), e(x-1,y+1)\right) \\ P(x,y) = arg\min\left(e(x-1,y-1), e(x-1,y), e(x-1,y+1)\right) \end{cases}, x! = 1$$

最终的值函数矩阵：

值函数矩阵：

0	0	127	255	383	510	510	510	510
0	0	127	255	383	510	510	510	510
127	127	254	255	383	510	510	637	637
128	255	255	510	383	383	511	511	638
128	255	255	255	255	383	383	510	510
127	127	127	255	383	255	382	382	382
0	0	127	255	383	510	255	255	255
0	0	127	255	383	510	510	255	255
0	0	127	255	383	510	510	510	255

最终的路径矩阵：

路径矩阵：

0	0	0	0	0	0	0	0	0
0	-1	-1	-1	-1	-1	-1	-1	-1
0	-1	-1	-1	-1	-1	1	-1	-1
0	-1	-1	-1	-1	1	0	1	1
0	1	1	1	0	0	1	1	1
0	1	1	0	-1	-1	0	1	1
0	0	0	-1	-1	-1	-1	0	0
0	-1	-1	-1	-1	-1	-1	-1	-1
0	-1	-1	-1	-1	-1	-1	-1	-1

⑤ 请问最佳的 x 方向的 seam 是什么？

有多条路径，从值函数矩阵最后一列中选择其中一个最小值，再根据路径矩阵推算最小seam，在原图中用红框标出：

原图像：

0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
127	127	127	127	255	127	127	127	127
255	255	255	255	255	255	255	255	255
127	127	127	127	255	127	127	127	127
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0

通过能量矩阵验证该结果，路径的总能量确实为255

能量矩阵：

0	0	127	128	128	127	0	0	0
0	0	127	128	128	127	0	0	0
127	127	254	128	128	127	127	127	127
128	128	128	256	128	128	128	128	128
128	128	128	128	0	128	128	128	128
127	127	127	128	128	0	127	127	127
0	0	127	128	128	127	0	0	0
0	0	127	128	128	127	0	0	0
0	0	127	128	128	127	0	0	0

⑥ 请问删除最佳 x 方向的 seam 得到的图像是什么？

不论选择哪条最短路径，删除后的图像相同：

0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
127	127	127	127	255	127	127	127	127
255	255	255	127	255	255	255	255	255
127	127	0	127	255	127	127	127	127
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0

⑦ 利用动态规划计算 y 方向的 Seam，请问初始化后的值函数矩阵和路径矩阵分别是多少；

初始条件：

$$\begin{cases} V(x,1) = e(x,1) \\ P(x,1) = 0 \end{cases}, x = 1 \dots 9$$

初始化的值函数矩阵：

0	0	127	128	128	127	0	0	0

初始化的路径矩阵：

0	0	0	0	0	0	0	0	0

⑧ 请问最终的值函数矩阵和路径矩阵分别是多少；

更新：

$$\begin{cases} V(x,y) = e(x,y) + \min\left(e(x-1,y-1), e(x,y-1), e(x+1,y-1)\right) \\ P(x,y) = arg\min\left(e(x-1,y-1), e(x,y-1), e(x+1,y-1)\right) \end{cases}, y! = 1$$

最终的值函数矩阵：

y向值函数矩阵：

0	0	127	128	128	127	0	0	0
0	0	127	255	255	127	0	0	0
127	127	254	255	255	127	127	127	127
255	255	255	510	255	255	255	255	255
383	383	383	383	255	383	383	383	383
510	510	510	383	383	255	510	510	510
510	510	510	511	383	382	255	510	510
510	510	637	511	510	382	255	255	510
510	510	637	638	510	382	255	255	255

最终的路径矩阵：

y向路径矩阵：

0	0	0	0	0	0	0	0	0
0	-1	-1	-1	1	1	0	-1	-1
0	-1	-1	-1	1	1	0	-1	-1
0	-1	-1	-1	1	0	-1	-1	-1
0	-1	-1	-1	0	-1	-1	-1	-1
0	-1	-1	1	0	-1	-1	-1	-1
0	-1	1	0	1	0	-1	-1	-1
0	-1	-1	1	1	1	0	-1	-1
0	-1	-1	1	1	1	0	-1	-1

⑨ 请问最佳的 y 方向的 seam 是什么？

有多条路径，从值函数矩阵最后一行中选择其中一个最小值，再根据路径矩阵推算最小seam，在中用红框标出：

原图像：

0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
127	127	127	127	255	127	127	127	127
255	255	255	255	255	255	255	255	255
127	127	127	127	255	127	127	127	127
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0

通过能量矩阵验证该结果，路径的总能量确实为255

能量矩阵：

0	0	127	128	128	127	0	0	0
0	0	127	128	128	127	0	0	0
127	127	254	128	128	127	127	127	127
128	128	128	256	128	128	128	128	128
128	128	128	128	0	128	128	128	128
127	127	127	128	128	0	127	127	127
0	0	127	128	128	127	0	0	0
0	0	127	128	128	127	0	0	0
0	0	127	128	128	127	0	0	0

⑩ 请问删除最佳 y 方向的 seam 得到的图像是什么？

不论选择哪条最短路径，删除后的图像相同：

0	0	0	127	255	127	0	0
0	0	0	127	255	127	0	0
0	0	0	127	255	0	0	0
127	127	127	127	127	127	127	127
255	255	255	255	255	255	255	255
127	127	127	127	255	127	127	127
0	0	0	127	255	127	0	0
0	0	0	127	255	127	0	0
0	0	0	127	255	127	0	0

若改用[-1,0,1]和[-1;0;1]计算梯度：

假设图像左上角为原点，水平方向为x轴，竖直方向为y轴，图像中的像素点从1开始计数，对边缘进行扩展，采用 0 值对超出边界的部分进行扩展。

① 它在 X 和 Y 方向的梯度分别是多少？

x方向梯度：

0	0	127	255	0	-255	-127	0	0
0	0	127	255	0	-255	-127	0	0
0	0	127	255	0	-255	-127	0	0
127	0	0	128	0	-128	0	0	-127
255	0	0	0	0	0	0	0	-255
127	0	0	128	0	-128	0	0	-127
0	0	127	255	0	-255	-127	0	0
0	0	127	255	0	-255	-127	0	0
0	0	127	255	0	-255	-127	0	0

y方向梯度：

0	0	0	127	255	127	0	0	0
0	0	0	0	0	0	0	0	0
127	127	127	0	0	0	127	127	127
255	255	255	128	0	128	255	255	255
0	0	0	0	0	0	0	0	0
-255	-255	-255	-128	0	-128	-255	-255	-255
-127	-127	-127	0	0	0	-127	-127	-127
0	0	0	0	0	0	0	0	0
0	0	0	-127	-255	-127	0	0	0

② 利用 L1 范数计算能量值，请问它的能量矩阵是多少？

能量矩阵：

0	0	127	382	255	382	127	0	0
0	0	127	255	0	255	127	0	0
127	127	254	255	0	255	254	127	127
382	255	255	256	0	256	255	255	382
255	0	0	0	0	0	0	0	255
382	255	255	256	0	256	255	255	382
127	127	254	255	0	255	254	127	127
0	0	127	255	0	255	127	0	0
0	0	127	382	255	382	127	0	0

③ 利用动态规划计算 x 方向的 Seam，请问初始化后的值函数矩阵和路径矩阵分别是多少；

初始化后的值函数矩阵：

0
0
127
382
255
382
127
0
0

初始化后的路径矩阵：

0
0
0
0
0
0
0
0
0

(4)请问最终的值函数矩阵和路径矩阵分别是多少；

最终的值函数矩阵：

值函数矩阵：

0	0	127	509	637	764	764	637	637
0	0	127	382	382	637	637	637	637
127	127	254	382	382	510	764	637	637
382	382	382	510	255	511	510	510	637
255	255	255	255	255	255	255	255	510
382	382	382	510	255	511	510	510	637
127	127	254	382	382	510	764	637	637
0	0	127	382	382	637	637	637	637
0	0	127	509	637	764	764	637	637

最终的路径矩阵：

路径矩阵：

0	0	0	0	1	1	1	1	0
0	0	0	0	0	0	1	0	0
0	-1	-1	-1	0	1	0	1	1
0	-1	-1	-1	1	0	1	1	1
0	0	0	0	0	0	0	0	0
0	1	1	1	-1	0	-1	-1	-1
0	1	1	1	0	-1	0	-1	-1
0	0	0	0	0	0	-1	0	0
0	0	0	0	-1	-1	-1	-1	0

(5)请问最佳的 x 方向的 seam 是什么？

从值函数矩阵最后一列中选择其中一个最小值，再根据路径矩阵推算最小seam，在原图中用红框标出：

原图像：

0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
127	127	127	127	255	127	127	127	127
255	255	255	255	255	255	255	255	255
127	127	127	127	255	127	127	127	127
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0

(6)请问删除最佳 x 方向的 seam 得到的图像是什么？

0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
127	127	127	127	255	127	127	127	127
127	127	127	127	255	127	127	127	127
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0

(7)利用动态规划计算 y 方向的 Seam，请问初始化后的值函数矩阵和路径矩阵分别是多少；

初始化的值函数矩阵：

0 0 127 382 255 382 127 0 0

初始化的路径矩阵：

0 0 0 0 0 0 0 0 0

(8)请问最终的值函数矩阵和路径矩阵分别是多少；

最终的值函数矩阵：

y向值函数矩阵：

0	0	127	382	255	382	127	0	0
0	0	127	382	255	382	127	0	0
127	127	254	382	255	382	254	127	127
509	382	382	510	255	510	382	382	509
637	382	382	255	255	255	382	382	637
764	637	510	511	255	511	510	637	764
764	637	764	510	255	510	764	637	764
637	637	637	510	255	510	637	637	637
637	637	637	637	510	637	637	637	637

最终的路径矩阵：

y向路径矩阵：

0	0	0	0	0	0	0	0	0
0	0	-1	-1	0	1	1	0	0
0	0	-1	-1	0	1	1	0	0
0	0	-1	-1	0	1	1	0	0
1	0	0	1	0	-1	0	0	-1
1	0	1	0	0	0	-1	0	-1
1	1	0	1	0	-1	0	-1	-1
1	0	1	1	0	-1	-1	0	-1
0	0	1	1	0	-1	-1	0	0

(9)请问最佳的 y 方向的 seam 是什么？

从值函数矩阵最后一行中选择其中一个最小值，再根据路径矩阵推算最小seam，在原图中用红框标出：

原图像：

0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
127	127	127	127	255	127	127	127	127
255	255	255	255	255	255	255	255	255
127	127	127	127	255	127	127	127	127
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0
0	0	0	127	255	127	0	0	0

(10)请问删除最佳 y 方向的 seam 得到的图像是什么？

0	0	0	127	127	0	0	0
0	0	0	127	127	0	0	0
0	0	0	127	127	0	0	0
127	127	127	127	127	127	127	127
255	255	255	255	255	255	255	255
127	127	127	127	127	127	127	127
0	0	0	127	127	0	0	0
0	0	0	127	127	0	0	0
0	0	0	127	127	0	0	0

实现完整的第二题所用的Matlab代码：

```
clc;clear;
SP = zeros(9);
for i = 1 : 1 : 9
    SP(4, i) = 127;
    SP(i, 4) = 127;
    SP(6, i) = 127;
    SP(i, 6) = 127;
end
for i = 1 : 1 : 9
    SP(5, i) = 255;
    SP(i, 5) = 255;
end
disp("原图像：");disp(SP);

%% Problem 1
% dx = zeros(9);
% dy = zeros(9);
% for i = 1 : 1 : 8
%     for j = 1 : 1 : 9
%         dx(j,i) = SP(j, i+1) - SP(j, i);
%     end
% end
% for i = 1 : 1 : 9
%     for j = 1 : 1 : 8
%         dy(j,i) = SP(j+1, i) - SP(j, i);
%     end
% end
% gradient_x = imfilter(dx, a);
% gradient_y = imfilter(dy, a);
%imfilter 采用的是模板运算
a = [1, 2, 1;
    2, 4, 2;
    1, 2, 1] / 16;
T = [-1, 0, 1];
T2 = [-1; 0; 1];

gradient_x = imfilter(SP, T);
gradient_y = imfilter(SP, T2);
disp("x 方向梯度：");disp(gradient_x);
disp("y 方向梯度：");disp(gradient_y);
dx = gradient_x;
dy = gradient_y;

%% Problem 2
E = abs(dx)+abs(dy);
Ecopy = E;
disp("能量矩阵：");disp(Ecopy);

%% Problem 3、4
my_max = 100000;
V = zeros(9);
P = zeros(9);
for i = 1 : 1 : 9
    for j = 1 : 1 : 9
        if(i == 1)
            choice = [0, 0];
        elseif(j == 1)
            choice = my_min([my_max; V(j, i-1); V(j+1, i-1)]);
        elseif(j == 9)
            choice = my_min([V(j-1, i-1); V(j, i-1); my_max]);
        else
            choice = my_min([V(j-1, i-1); V(j, i-1); V(j+1, i-1)]);
        end
        V(j, i) = E(j, i) + choice(1);
        P(j, i) = choice(2);
    end
end
Vcopy = V;
Pcopy = P;
```

```
disp("值函数矩阵：");disp(Vcopy);
disp("路径矩阵：");disp(Pcopy);

%% Problem 5、6
my_max = 100000;
V = zeros(9);
P = zeros(9);
%直接转置 E，其余操作与 3、4 问相同
E = E';
for i = 1 : 1 : 9
    for j = 1 : 1 : 9
        if(i == 1)
            choice = [0, 0];
        elseif(j == 1)
            choice = my_min([my_max; V(j, i-1); V(j+1, i-1)]);
        elseif(j == 9)
            choice = my_min([V(j-1, i-1); V(j, i-1); my_max]);
        else
            choice = my_min([V(j-1, i-1); V(j, i-1); V(j+1, i-1)]);
        end
        V(j, i) = E(j, i) + choice(1);
        P(j, i) = choice(2);
    end
end
V = V';
P = P';
disp("y 向值函数矩阵：");disp(V);
disp("y 向路径矩阵：");disp(P);

function s=my_min(n)
    if(n(1)<n(2) && n(1)<n(3))
        s = [n(1), -1];
    elseif(n(2)<=n(1) && n(2)<=n(3))
        s = [n(2), 0];
    else
        s = [n(3), 1];
    end
end
```