$Q'[s, a] = (1 - \alpha) \cdot Q[s, a] + \alpha \cdot (r + \gamma \cdot Q[s', \mathrm{argmax}_{a'}(Q[s', a'])])$

if all actions at a given state are the same value, randomly choose an action to proceed.

small world	
-1	-1
-100	1

small world	
s0	s1
s2	s3

a0 = up
a1 = down
a2 = left
a3 = right

Step 0: initial Q Table and initialize robot at s0

Q Table				
	a0	**a1**	**a2**	**a3**
s0	0	0	0	0
s1	0	0	0	0
s2	0	0	0	0
s3	0	0	0	0

At this point, here is what your robot believe is the best policy base on Q Table

small world	
<, ^,v or >	<, ^,v or >
<, ^,v or >	<,^, v or >

Step 1: Robot takes a step right, going from s0 to s1

$Q'[0, 3] = (1 - .5) \cdot Q[0, 3] + .5 \cdot (-1 + .9 \cdot Q[1, \mathrm{argmax}_{a'}(Q[1, :])]) = -.5$

Q Table				
	a0	**a1**	**a2**	**a3**
s0	0	0	0	**-0.5**
s1	0	0	0	0
s2	0	0	0	0
s3	0	0	0	0

At this point, here is what your robot believe is the best policy base on Q Table

small world	
<, ^, or v	<, ^,v or >
<, ^,v or >	<,^, v or >

Step 2: Robot takes a step left, going from s1 to s0

$Q'[1, 2] = (1 - .5) \cdot Q[1, 2] + .5 \cdot (-1 + .9 \cdot Q[0, \mathrm{argmax}_{a'}(Q[0, :])]) = -.5$

Q Table				
	a0	**a1**	**a2**	**a3**
s0	0	0	0	-0.5
s1	0	0	**-0.5**	0
s2	0	0	0	0
s3	0	0	0	0

At this point, here is what your robot believe is the best policy base on Q Table

small world	
<, ^, or v	^, v or >
<, ^,v or >	<,^, v or >

Step 3: Robot takes a step down going from s0 to s2

$Q'[0, 1] = (1 - .5) \cdot Q[0, 1] + .5 \cdot (-100 + .9 \cdot Q[1, \mathrm{argmax}_{a'}(Q[2, :])]) = -.5$

Q Table				
	a0	**a1**	**a2**	**a3**
s0	0	**-50**	0	-0.5
s1	0	0	-0.5	0
s2	0	0	0	0
s3	0	0	0	0

At this point, here is what your robot believe is the best policy base on Q Table

small world	
< or >	^, v or >
<, ^,v or >	<,^, v or >

Step 4: Robot takes a step down going from s2 to s3

$Q'[2, 3] = (1 - .5) \cdot Q[2, 3] + .5 \cdot (1 + .9 \cdot Q[1, \mathrm{argmax}_{a'}(Q[3, :])]) = -.5$

^ red goes to 0 because terminal state has no future action

Q Table				
	a0	**a1**	**a2**	**a3**
s0	0	-50	0	-0.5
s1	0	0	-0.5	0
s2	0	0	0	**0.5**
s3	0	0	0	0

At this point, here is what your robot believe is the best policy base on Q Table

small world	
< or >	^, v or >
>	<,^, v or >

After it reached terminal state, the robot restarts at s0, but the Q Table remains

...Many steps later:

Q Table				
	a0	**a1**	**a2**	**a3**
s0	-32	-80	-35	59
s1	-15	90	-42	-30
s2	-51	-17	-30	90
s3	0	0	0	0

By now, your robot knows to avoid quicksand and moves towards the goal

small world	
>	v
>	<,^, v or >

Your Q Table should look something similar to this (the numbers are not accurate, just for demonstration purpose), in which the largest value of a row will represent the action your robots will take.