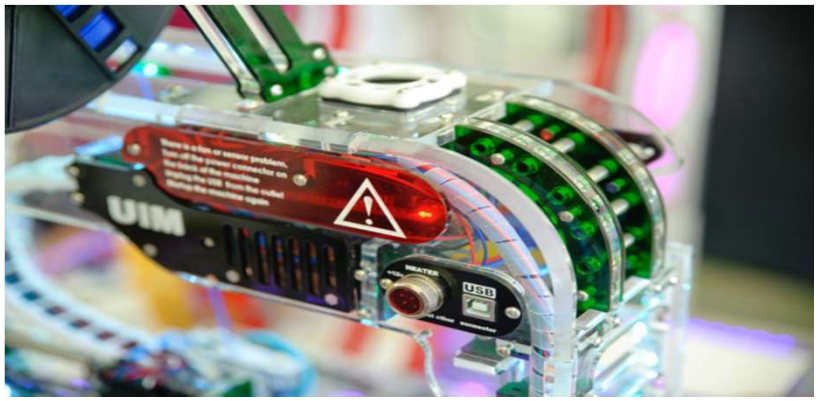
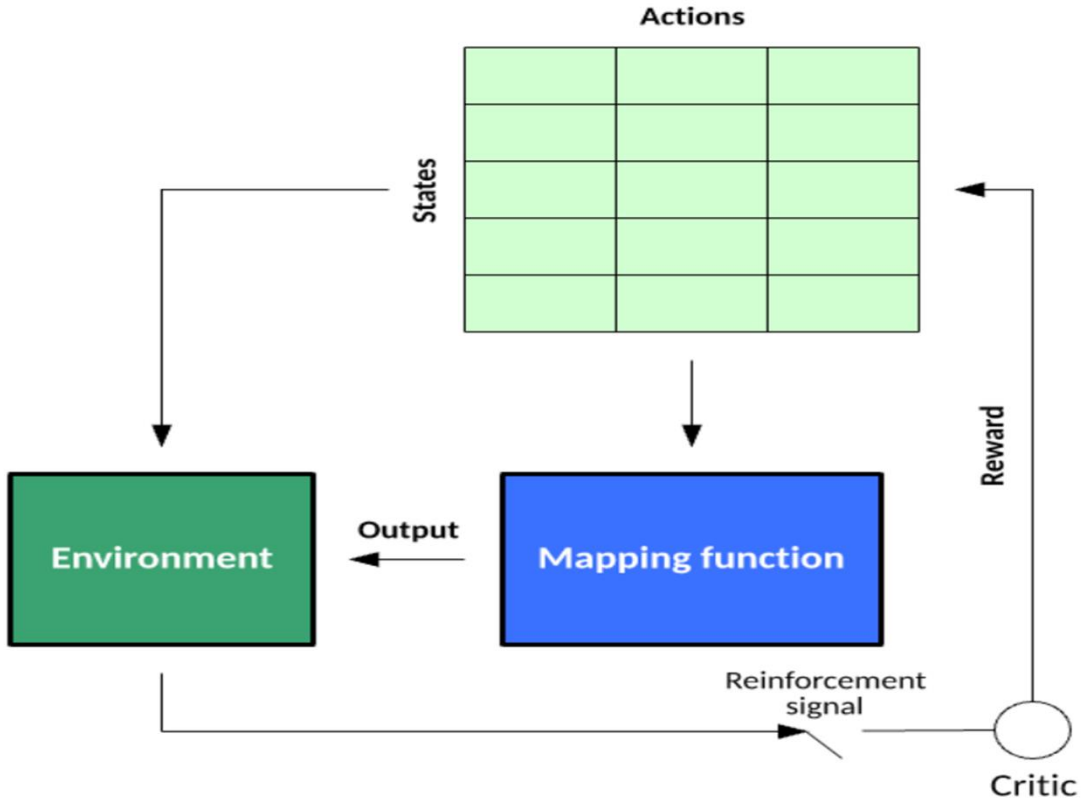
Machine Learning: Aprendizaje por refuerzo
Algoritmo Q



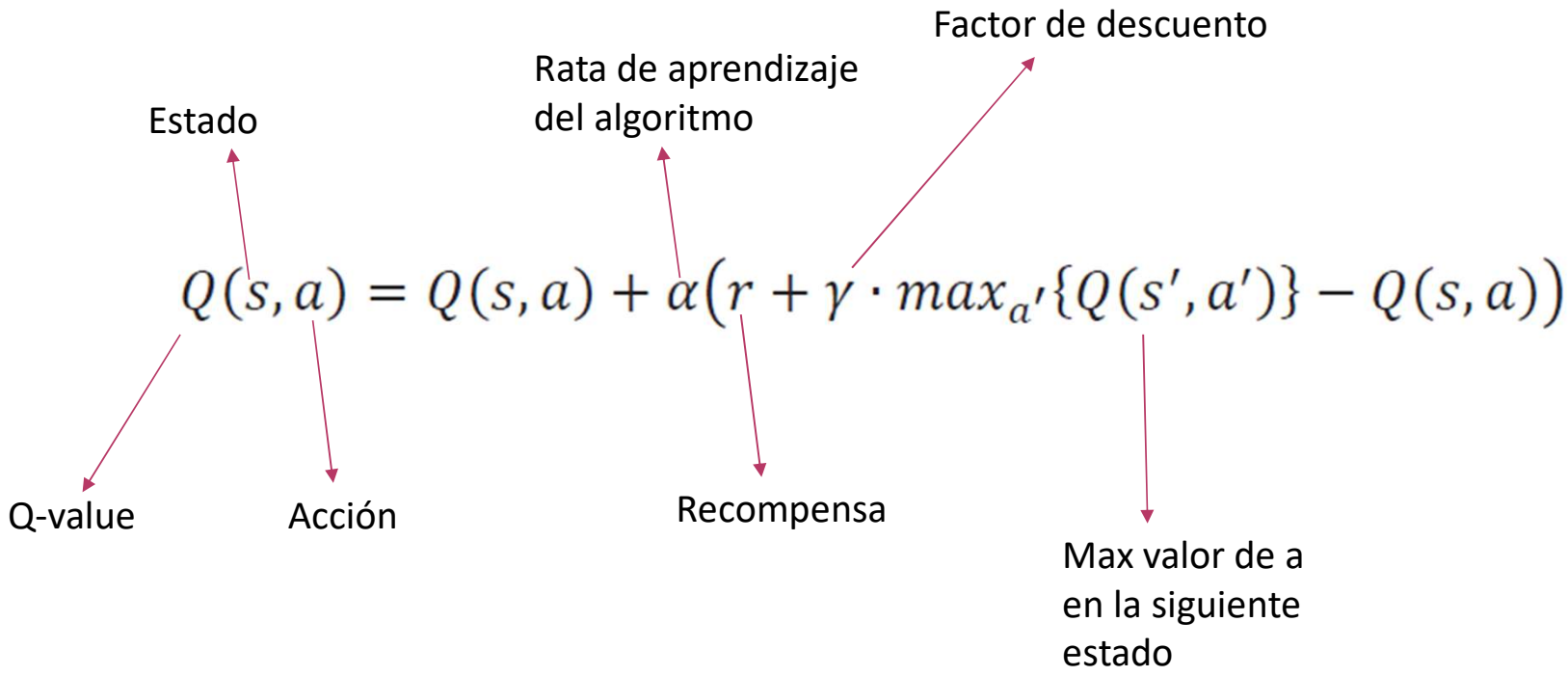
Aprendizaje por refuerzo- Algoritmo Q

Es un paradigma que puede ser usado para que un agente, encuentre una política (Aprendizaje) optima para escoger la acción a desarrollar en una proceso de Márkov.

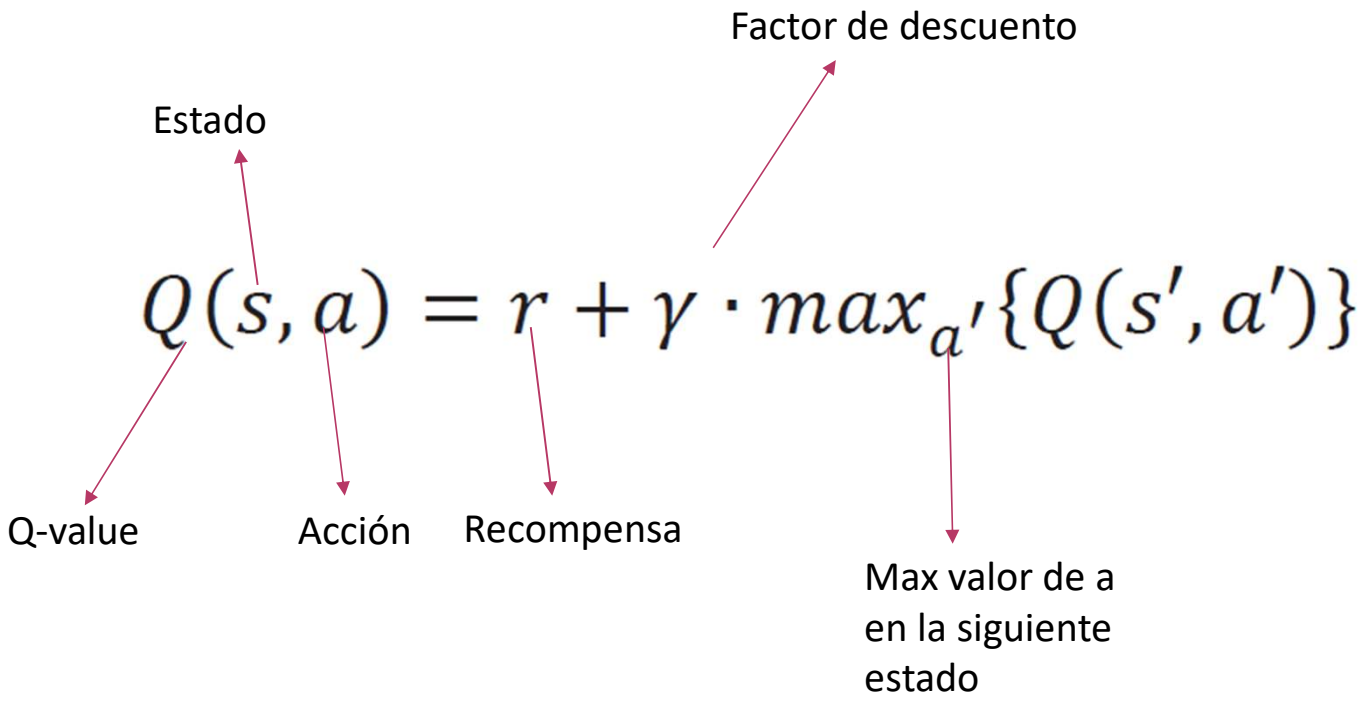
A. A. Марков (1886).



Algoritmo de aprendizaje Q



Algoritmo de aprendizaje Q



Algoritmo de aprendizaje Q

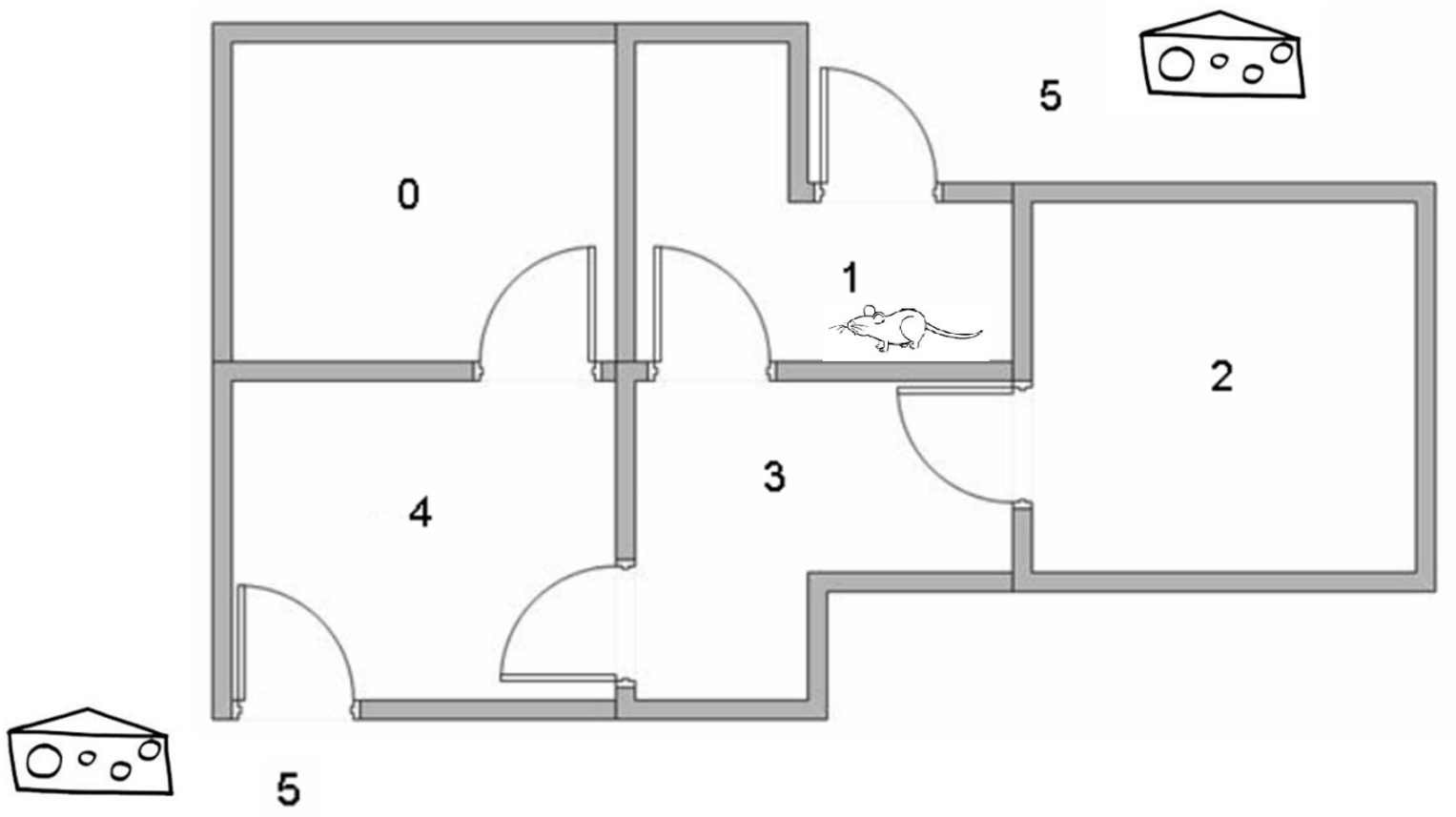
$$Q(s,a) = r + \gamma \cdot max_{a'}\{Q(s',a')\}$$

1. Seleccione gamma (factor de descuento) y obtenga matriz de recompensa R.
2. Inicializar matriz Q, todo en 0.
3. Para cada corrida:
   Seleccione aleatoriamente un estado.
   Do While mientras no se consiga el estado final.
   - •Seleccione una entre todas las acciones posibles para el estado actual.
   - •Usando esa posible condición, considere ir al siguiente estado
   - •Tome el valor Max Q de ese siguiente estado de todas las posibilidades.
   - •Calcule: $Q(s,a) = r + \gamma \cdot max_{a'}\{Q(s',a')\}$
   - •Seleccione el siguiente estado como el estado actual

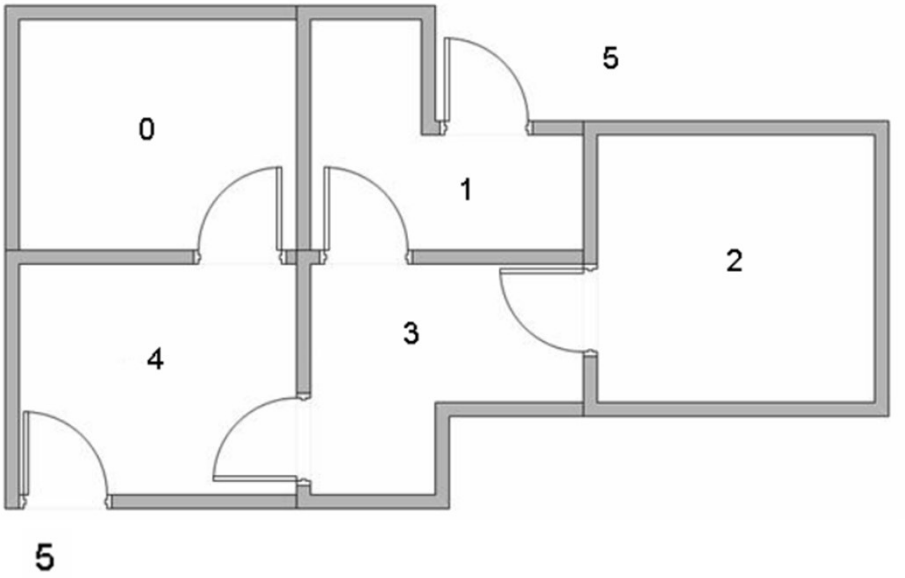
   End Do

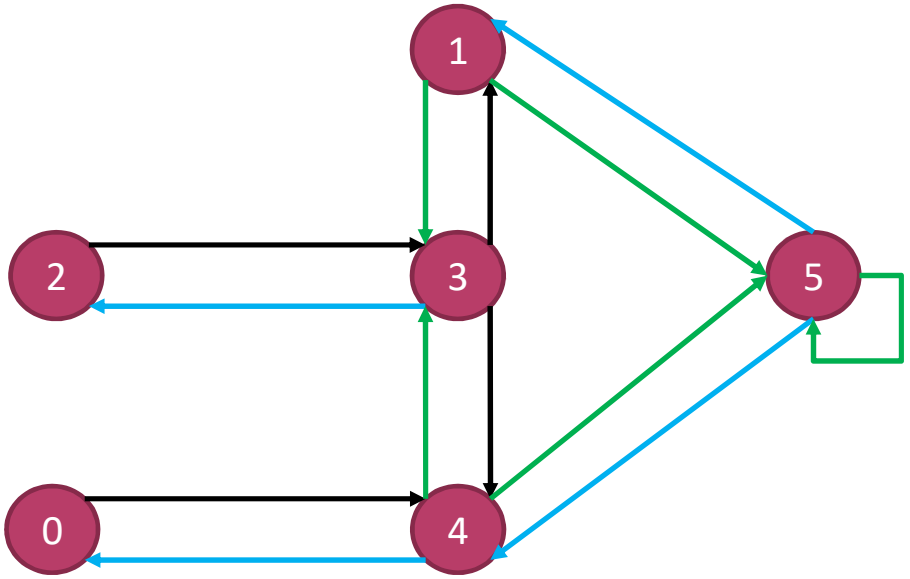
End For

Algoritmo de aprendizaje Q



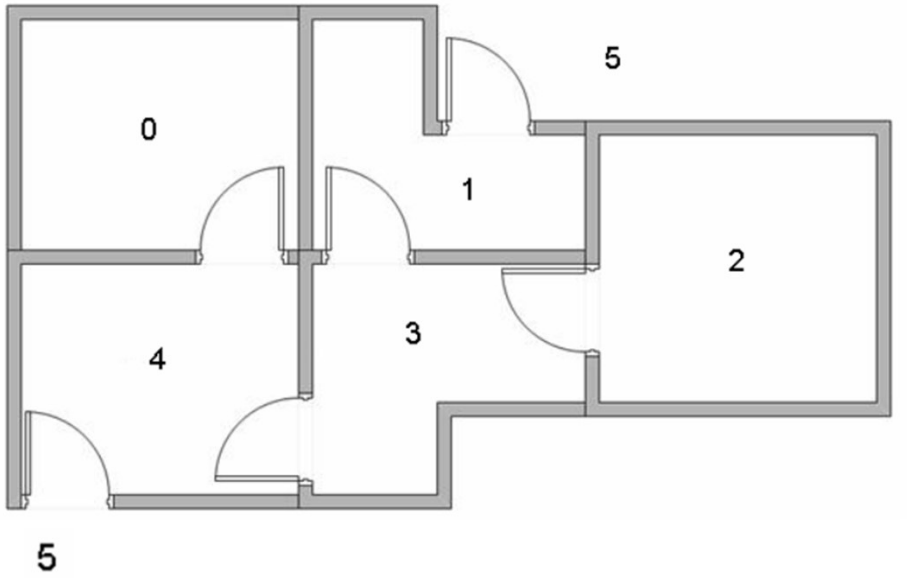
Algoritmo de aprendizaje Q

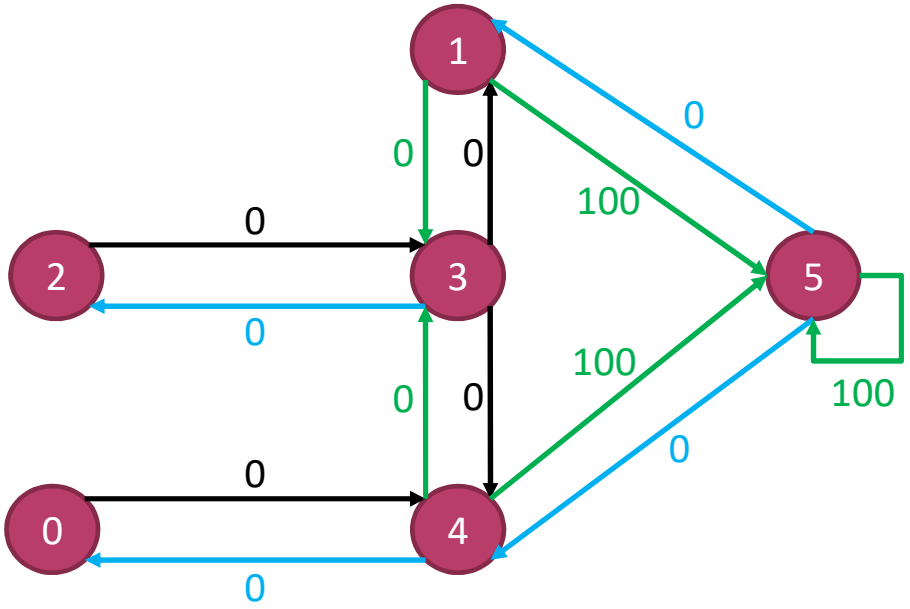




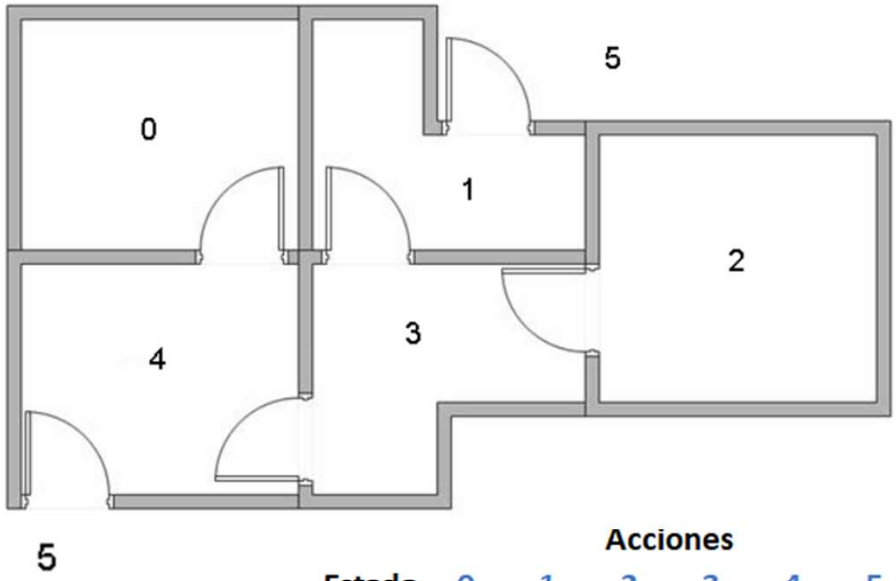
Gamma = 0.8
Obtener matriz de recompensas.

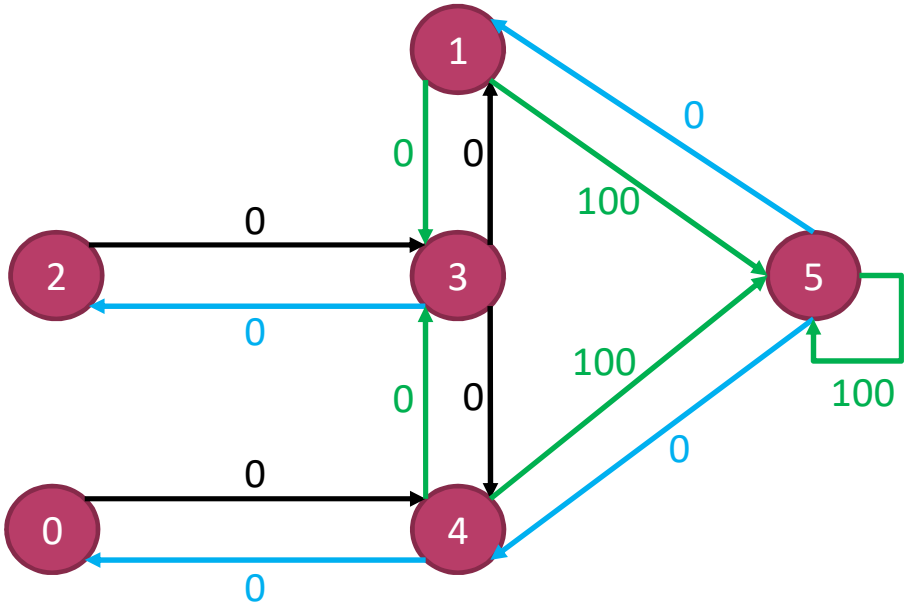
Algoritmo de aprendizaje Q





Algoritmo de aprendizaje Q





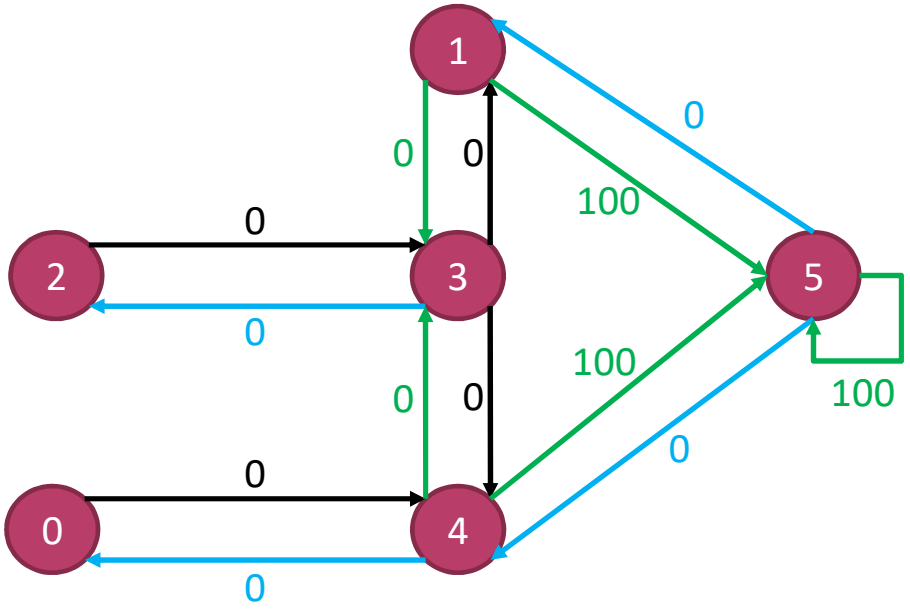
$R=$

Estado	Acciones 0	1	2	3	4	5
0	-1	-1	-1	-1	0	-1
1	-1	-1	-1	0	-1	100
2	-1	-1	-1	0	-1	-1
3	-1	0	0	-1	0	-1
4	0	-1	-1	0	-1	100
5	-1	0	-1	-1	0	100

Algoritmo de aprendizaje Q

$Q=$

Estado	Acciones 0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0



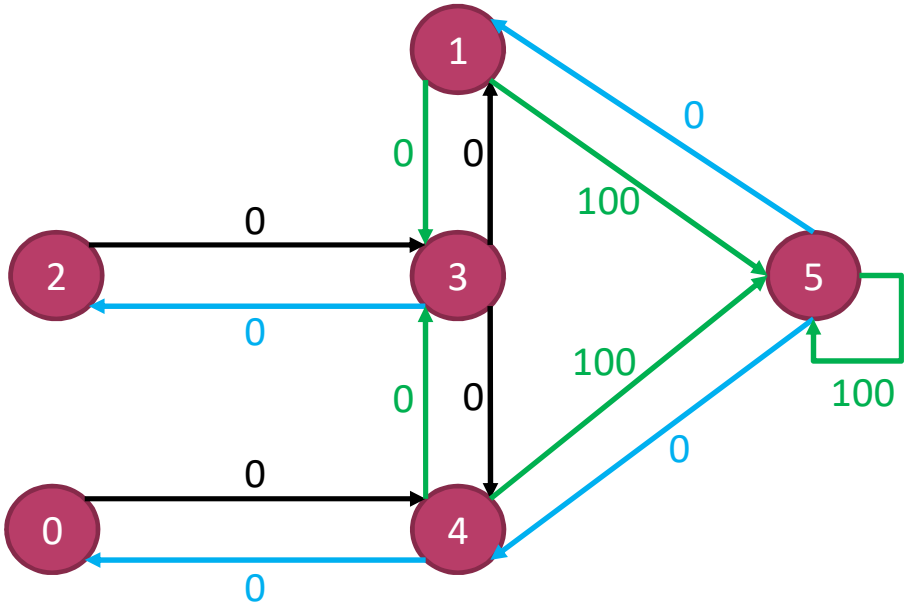
Algoritmo de aprendizaje Q

$R=$

Estado	Acciones 0	1	2	3	4	5
0	-1	-1	-1	-1	0	-1
1	-1	-1	-1	0	-1	100
2	-1	-1	-1	0	-1	-1
3	-1	0	0	-1	0	-1
4	0	-1	-1	0	-1	100
5	-1	0	-1	-1	0	100

$Q=$

Estado	Acciones 0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0



Aleatorio estado 1.

Aleatorio estado 1 pasa a estado 5.

En el estado 5, que pasaria?

Q(estado, accion) = R(estado, accion) + Gamma * Max[Q(sig_estado, todas las acciones)]
Q(1, 5) = R(1, 5) + 0.8 * Max[Q(5, 1), Q(5, 4), Q(5, 5)] =
Q(1, 5) = 100 + 0.8 * 0 = 100
Q(1, 5)=100

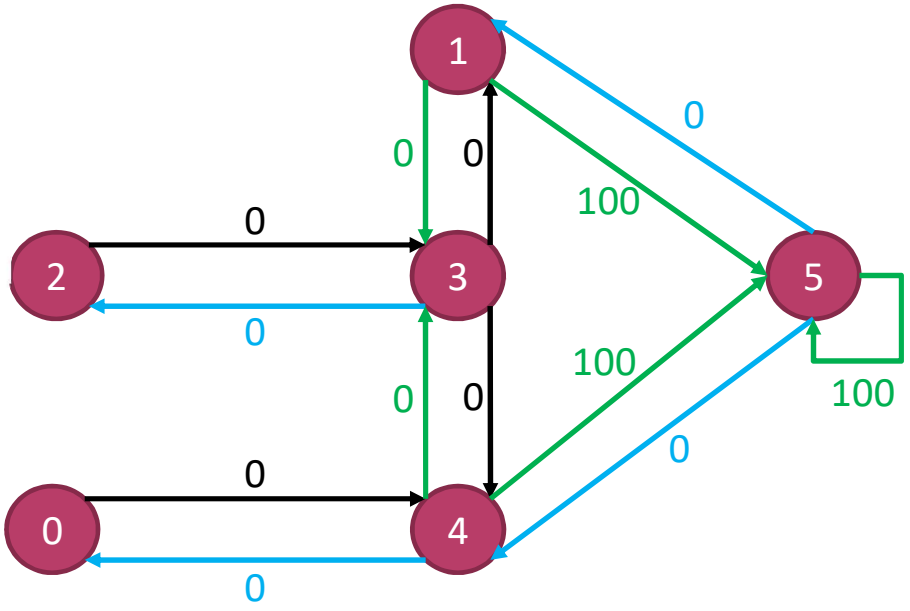
Algoritmo de aprendizaje Q

$R=$

Estado	Acciones 0	1	2	3	4	5
0	-1	-1	-1	-1	0	-1
1	-1	-1	-1	0	-1	100
2	-1	-1	-1	0	-1	-1
3	-1	0	0	-1	0	-1
4	0	-1	-1	0	-1	100
5	-1	0	-1	-1	0	100

$Q=$

Estado	Acciones 0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	0	100
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0



Aleatorio estado 1.

Aleatorio estado 1 pasa a estado 5.

En el estado 5, que pasaria?

Q(estado, accion) = R(estado, accion) + Gamma * Max[Q(sig_estado, todas las acciones)]
Q(1, 5) = R(1, 5) + 0.8 * Max[Q(5, 1), Q(5, 4), Q(5, 5)] =
Q(1, 5) = 100 + 0.8 * 0 = 100
Q(1, 5)=100

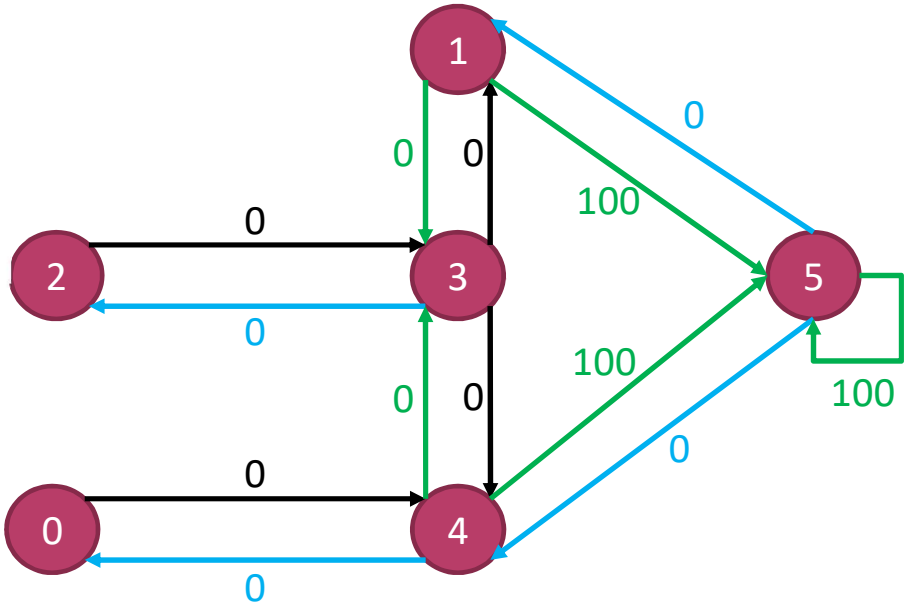
Algoritmo de aprendizaje Q

$R=$

Estado	Acciones 0	1	2	3	4	5
0	-1	-1	-1	-1	0	-1
1	-1	-1	-1	0	-1	100
2	-1	-1	-1	0	-1	-1
3	-1	0	0	-1	0	-1
4	0	-1	-1	0	-1	100
5	-1	0	-1	-1	0	100

$Q=$

Estado	Acciones 0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	0	100
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0



Aleatorio estado 3.

Aleatorio estado 3 pasa a estado 1.

En el estado 1, que pasaria?

Q(estado, accion) = R(estado, accion) + Gamma * Max[Q(sig_estado, todas las acciones)]
Q(3, 1) = R(3, 1) + 0.8 * Max[Q(1, 3), Q(1, 5)] = 0 + 0.8 * Max(0, 100) = 80
Q(3,1) = 0 + 0.8 * 100 = 80
Q(3,1)=80

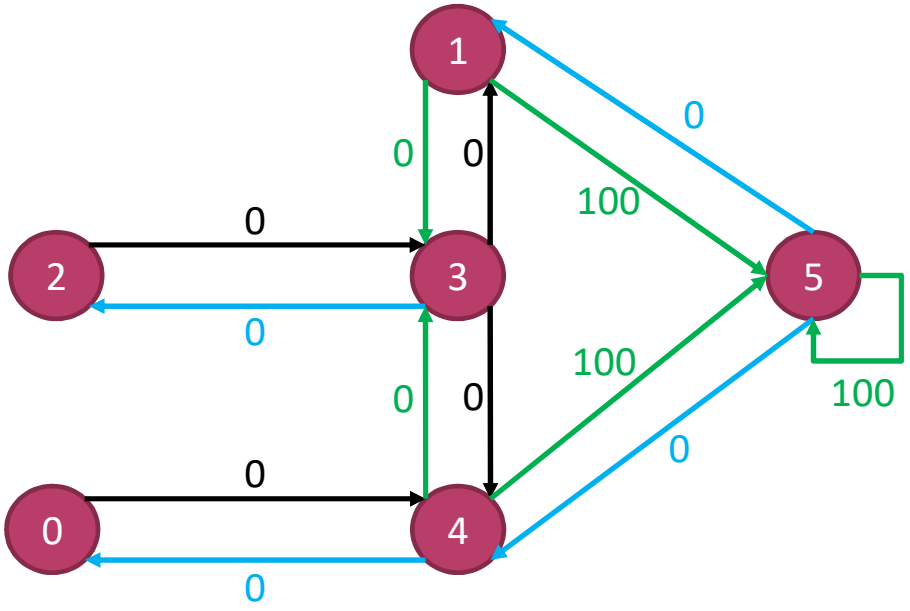
Algoritmo de aprendizaje Q

$R=$

Estado	Acciones 0	1	2	3	4	5
0	-1	-1	-1	-1	0	-1
1	-1	-1	-1	0	-1	100
2	-1	-1	-1	0	-1	-1
3	-1	0	0	-1	0	-1
4	0	-1	-1	0	-1	100
5	-1	0	-1	-1	0	100

$Q=$

Estado	Acciones 0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	0	100
2	0	0	0	0	0	0
3	0	80	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0



Aleatorio estado 3.

Aleatorio estado 3 pasa a estado 1.

En el estado 1, que pasaría?

Q(estado, accion) = R(estado, accion) + Gamma * Max[Q(sig_estado, todas las acciones)]
Q(3, 1) = R(3, 1) + 0.8 * Max[Q(1, 2), Q(1, 5)] = 0 + 0.8 * Max(0, 100)
Q(3,1) = 0 + 0.8 * 100 = 80
Q(3,1)=80

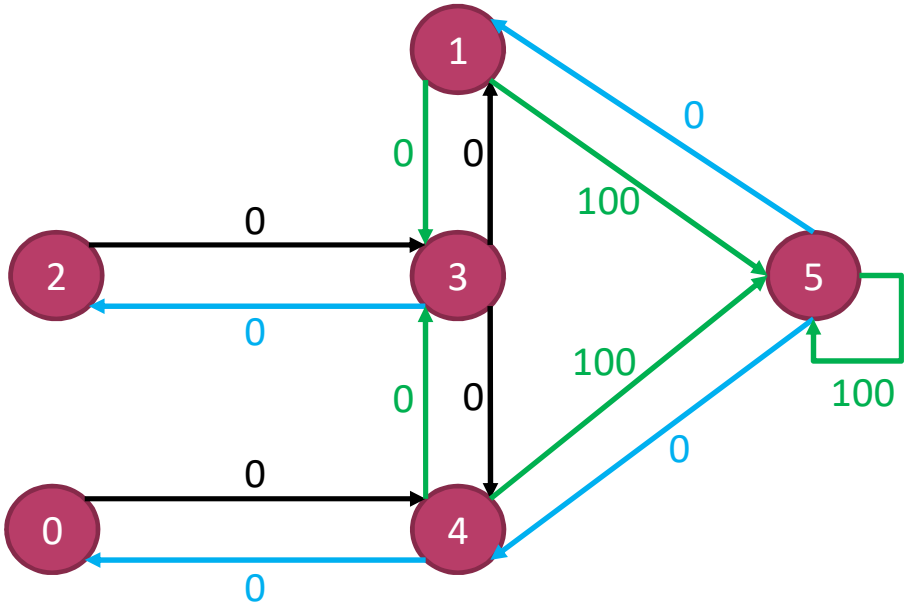
Algoritmo de aprendizaje Q

$R=$

Estado	Acciones 0	1	2	3	4	5
0	-1	-1	-1	-1	0	-1
1	-1	-1	-1	0	-1	100
2	-1	-1	-1	0	-1	-1
3	-1	0	0	-1	0	-1
4	0	-1	-1	0	-1	100
5	-1	0	-1	-1	0	100

$Q=$

Estado	Acciones 0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	0	100
2	0	0	0	0	0	0
3	0	80	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0



Estado 1.

Estado 1 pasa a estado 5.

En el estado 5, que pasaría?

Q(estado, accion) = R(estado, accion) + Gamma * Max[Q(sig_estado, todas las acciones)]
Q(1, 5) = R(1, 5) + 0.8 * Max[Q(5, 1), Q(5, 4), Q(5, 5)] = 100 + 0.8 * 0 = 100
Q(1,5) = 100 + 0.8 * 0 = 100
Q(1,5)=100

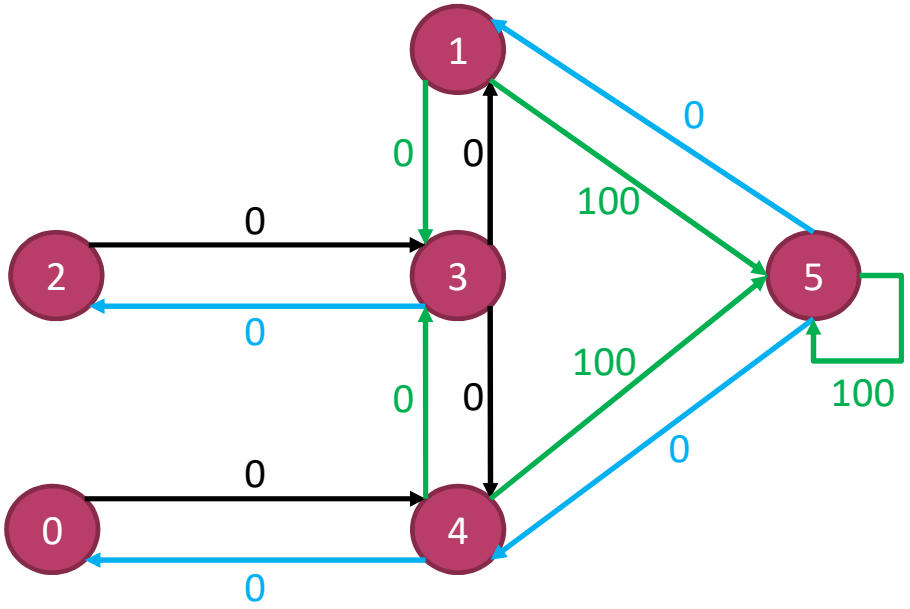
Algoritmo de aprendizaje Q

$Q=$

Estado	Acciones 0	1	2	3	4	5
0	0	0	0	0	400	0
1	0	0	0	320	0	500
2	0	0	0	320	0	0
3	0	400	256	0	400	0
4	320	0	0	320	0	500
5	0	400	0	0	400	500

$Q=$

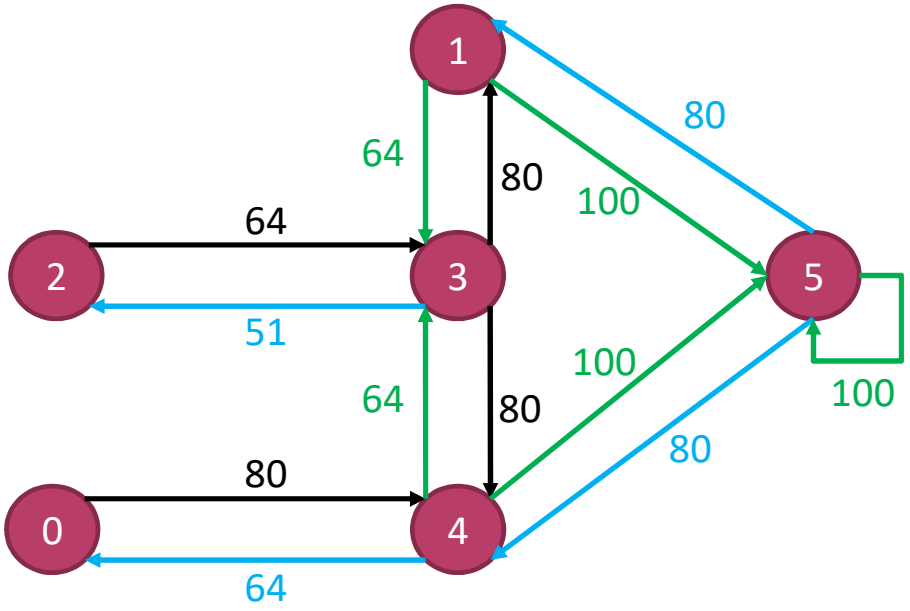
Estado	Acciones 0	1	2	3	4	5
0	0	0	0	0	80	0
1	0	0	0	64	0	100
2	0	0	0	64	0	0
3	0	80	51	0	80	0
4	64	0	0	64	0	100
5	0	80	0	0	80	100



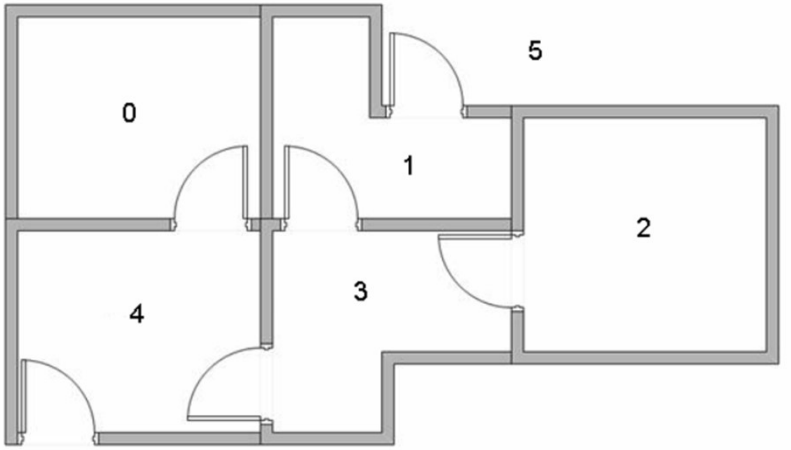
Algoritmo de aprendizaje Q

$Q=$

Estado	Acciones 0	1	2	3	4	5
0	0	0	0	0	80	0
1	0	0	0	64	0	100
2	0	0	0	64	0	0
3	0	80	51	0	80	0
4	64	0	0	64	0	100
5	0	80	0	0	80	100

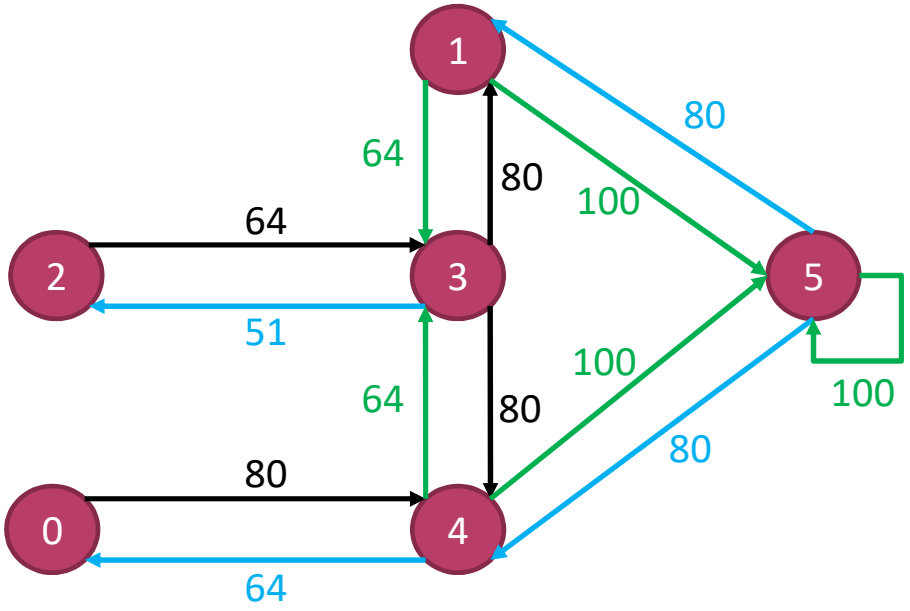


Algoritmo de aprendizaje Q



$Q=$

Estado	Acciones 0	1	2	3	4	5
0	0	0	0	0	80	0
1	0	0	0	64	0	100
2	0	0	0	64	0	0
3	0	80	51	0	80	0
4	64	0	0	64	0	100
5	0	80	0	0	80	100



Ya aprendiste una técnica de ML->
REINFORCEMENT LEARNING Q

Fin de esta sección.
Pon a prueba tu conocimiento adquirido.