Métodos de Monte Carlo

En este ejercicio vamos a implementar la primera solución para los problemas de aprendizaje por refuerzo, los métodos de Monte Carlo.

Recuerde que el método de Monte Carlo consiste en la colección de muestras calculando los valores para la secuencia completa de los estados hasta el estado final. Una vez se han coleccionado "suficientes" muestras, el valor de los estados se toma como el valor promedio de las muestras sobre las cuales apareció el estado.

Para resolver problemas de aprendizaje por refuerzo utilizando el método de Monte Carlo crearemos un archivo `mcm.py`. Inicialmente utilizaremos este archivo para solucionar el problema sobre el ambiente de Gridworld (suponiendo un ruido de `0.25` para las acciones, es decir que la probabilidad de ejecutar la acción deseada es de 0.75 y una acción aleatoria (dividida en partes iguales) con probabilidad de 0.25).

Task 1

1. Implemente la classe `MCM` para solucionar Gridworld sin conocer los detalles del modelo de MDP para el problema. Es decir, en este caso, nuestro agente de `MCM` no tendrá acceso al `mdp` como era el caso para la iteración de valores o iteración de políticas.

2. El comportamiento del agente (de Monte Carlo) esta dado por dos momentos. El proceso de recolección de muestras y el proceso de explotación de las mismas, es decir, el cálculo de la política del agente. Usted debe implementar el comportamiento del agente dado que, ejecutando episodios como muestras, sea capaz de calcular los valores para los estados.

Para la implementación de `MCM` responda las siguientes preguntas. Tenga en cuenta que debe ejecutar su agente múltiples veces para poder observar el comportamiento (una sola instancia no nos puede llevar a ninguna conclución). Justifique su respuestas con análisis de la ejejcución y gráficas del comportamiento.

1. ¿Cuantas muestras debe tomar el agente? Su implementación no debe utilizar este número como un parámetro o tenerlo como un factor predeterminado del agente.
2. ¿Cómo se comparan los valores de `MCM` con respecto a los valores obtenidos en el ejercicio de iteración de valores `value_iteration`? ¿Porqué se da la diferencia si existe alguna, o porqué no existe ninguna diferencia?
3. ¿Cómo se compara la política obtenida utilizando `MCM` con respecto a la política obtenida en el jercicio de iteración de políticas `policy_iteration`? ¿Porqué se da la diferencia si existe alguna, o porqué no existe ninguna diferencia?

4. ¿Cuál es el efecto de del factor de descuento sobre el método de Monte Carlo, calcule la solución de Gridworld con diferentes valores?

In [5]:
```
from environment_world import EnvironmentWorld
from mcm import MonteCarloAgent

grid_world = EnvironmentWorld([
    ['S'] + [' '] * 9,
    [' '] * 10,
    [' ', '#', '#', '#', '#', ' ', '#', '#', '#', ' '],
    [' ', ' ', ' ', ' ', '#', ' ', ' ', ' ', ' ', ' '],
    [' ', ' ', ' ', ' ', '#', '-1', ' ', ' ', ' ', ' '],
    [' ', ' ', ' ', ' ', '#', '+1', ' ', ' ', ' ', ' '],
    [' ', ' ', ' ', ' ', '#', ' ', ' ', ' ', ' ', ' '],
    [' ', ' ', ' ', ' ', '#', '-1', '-1', ' ', ' ', ' '],
    [' '] * 10,
    [' '] * 10
], noise=0.25)
```

In [2]:
```
montecarlo_agent = MonteCarloAgent(grid_world, discount_factor=0.9, initial_
montecarlo_agent.learn(max_episodes=1000, convergence_check_frequency=10000,
montecarlo_agent.print_policy()
```

```
Episodes: 100%|██████████| 1000/1000 [10:32<00:00,  1.58it/s]
       0      1      2      3      4      5      6      7      8     9
0   down  right  right  right  right  right  right  right  right  down
1  right     up   left   left   left   left     up  right     up  down
2     up   None   None   None   None     up   None   None   None  down
3     up     up  right     up   None     up  right   down   left  down
4     up   left   left   left   None   None   down   down   down  left
5     up   left     up   left   None   None   left   left   left    up
6     up   left     up     up   None     up     up     up     up    up
7     up   left   left     up   None   None   None  right  right    up
8     up   left   left   left   left   left   down  right  right  left
9   left     up   left   left   left   left  right  right  right    up
```

In [3]:
```
montecarlo_agent = MonteCarloAgent(grid_world, discount_factor=0.5, initial_
montecarlo_agent.learn(max_episodes=1000, convergence_check_frequency=10, cc
montecarlo_agent.print_policy()
```

```
Episodes: 100%|██████████| 1000/1000 [00:47<00:00, 20.98it/s]
      0     1     2     3     4      5      6      7      8      9
0  down    up  left    up    up  right  right  right   down   down
1    up  down    up    up  left   left   down  right  right   down
2  left  None  None  None  None     up   None   None   None   down
3  left  down    up  left  None     up  right  right   down   down
4  left    up  left  left  None   None   down   down   down   down
5    up  left  left    up  None   None   left   left   left   left
6    up  left  left    up  None     up     up     up     up     up
7  left    up  left    up  None   None   None  right  right     up
8    up  down    up    up  down   down   down   down  right   down
9  down  left  left  left  left   left  right  right   down  right
```

In [4]:
```
montecarlo_agent = MonteCarloAgent(grid_world, discount_factor=1, initial_ep
montecarlo_agent.learn(max_episodes=1000, convergence_check_frequency=10, cc
montecarlo_agent.print_policy()
```

```
Episodes: 100%|██████████| 1000/1000 [00:05<00:00, 181.80it/s]
```

	0	1	2	3	4	5	6	7	8	9
0	down	left	right	left	down	down	left	up	right	down
1	right	right	left	up	right	right	left	down	down	down
2	down	None	None	None	None	left	None	None	None	down
3	up	left	up	down	None	right	down	down	left	right
4	up	down	right	left	None	None	down	down	right	down
5	up	down	up	down	None	None	left	left	right	left
6	up	left	up	down	None	up	right	up	left	right
7	right	down	right	left	None	None	None	up	left	up
8	left	up	right	down	left	left	right	down	up	left
9	up	down	up	left	left	up	left	up	right	up

In [10]:
```
def print_value_iteration(value_iteration, previous_values=None, previous_it
    print(f'\n{"_" * 8} iterations={value_iteration.iterations} discount={va
    current_value = pd.DataFrame(value_iteration.values)
    if previous_values is not None:
        average_square_error = ((current_value - previous_values) ** 2).fill
                current_value.count().sum() * (value_iteration.iterations -
        print(f'average_square_error (measure of convergence) = {average_squ
    print("___values___")
    print(value_iteration)
    print("___policy___")
    print(value_iteration.get_full_policy().map(lambda action: action.name i
```

In [11]:
```
from assignment_montecarlo.value_iteration import ValueIteration
import pandas as pd

previous_values = None
previous_iterations = 0
grid_value_iteration = ValueIteration(grid_world, discount=0.9)
for iteration_number in [10] * 10:
    grid_value_iteration.run_value_iteration(iterations=iteration_number)
    print_value_iteration(grid_value_iteration, previous_values, previous_it
    previous_values = pd.DataFrame(grid_value_iteration.values)
    previous_iterations += iteration_number
```

```
_________ iterations=100 discount=0.9 _________
___values___
              0             1             2             3         4
5  \
0 -1.259558e-10 -1.357687e-09  9.394524e-02  1.299479e-01  0.223583  0.24664
8
1 -2.719390e-10  7.633054e-02  1.126215e-01  2.135490e-01  0.245981  0.32006
4
2 -1.295890e-11  0.000000e+00  0.000000e+00  0.000000e+00  0.000000  0.36279
9
3 -6.694781e-13 -1.405904e-11 -6.034699e-11 -3.134467e-10  0.000000  0.42448
7
4 -2.343173e-11 -1.274357e-10 -1.172151e-09 -4.079587e-09  0.000000  0.00000
0
5 -1.855355e-10 -2.310449e-09 -1.407781e-08 -6.738875e-08  0.000000  0.00000
0
6 -2.504977e-09 -2.038024e-08 -1.847121e-07 -1.051710e-06  0.000000  0.83679
1
7 -1.528886e-08 -1.814655e-07 -1.958662e-06  4.606123e-02  0.000000  0.00000
0
8 -7.450628e-08 -9.685962e-07  4.606192e-02  4.701256e-02  0.138915  0.10182
8
9 -6.377817e-08 -3.665784e-07 -4.481108e-06  1.055245e-01  0.126061  0.25967
9

          6         7         8         9
0  0.223583  0.129948  0.145036  0.198973
1  0.245981  0.217585  0.198180  0.334030
2  0.000000  0.000000  0.000000  0.388348
3  0.597247  0.611214  0.523482  0.484365
4  0.696432  0.688406  0.614101  0.521063
5  0.935996  0.803205  0.681450  0.604149
6  0.709170  0.681856  0.610044  0.513585
7  0.000000  0.501532  0.498301  0.481206
8  0.289380  0.397630  0.468331  0.381785
9  0.277723  0.383398  0.349691  0.360074
___policy___
       0      1      2      3      4      5      6     7      8     9
0     UP  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  LEFT  RIGHT  DOWN
1  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  LEFT  RIGHT  DOWN
2   DOWN   NONE   NONE   NONE   NONE   DOWN   NONE  NONE   NONE  DOWN
3   LEFT   LEFT   LEFT   LEFT   NONE  RIGHT   DOWN  DOWN   DOWN  LEFT
4     UP     UP     UP     UP   NONE   NONE   DOWN  DOWN   LEFT  LEFT
5     UP     UP     UP     UP   NONE   NONE   LEFT  LEFT   LEFT  LEFT
6     UP     UP     UP   DOWN   NONE     UP     UP    UP   LEFT  LEFT
7     UP   LEFT   DOWN   DOWN   NONE   NONE   NONE    UP     UP    UP
8     UP  RIGHT  RIGHT  RIGHT     UP  RIGHT  RIGHT    UP     UP    UP
9   DOWN   LEFT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT    UP     UP    UP

_________ iterations=100 discount=0.9 _________
average_square_error (measure of convergence) = 5.6819305162854295e-05
___values___
          0         1         2         3         4         5         6  \
0  0.142483  0.161342  0.184633  0.210063  0.239216  0.271586  0.242359
1  0.151810  0.176157  0.203112  0.234825  0.271143  0.313629  0.273262
2  0.131488  0.000000  0.000000  0.000000  0.000000  0.366056  0.000000
```

```
3  0.109645  0.090968  0.071466  0.070189  0.000000  0.420180  0.599574
4  0.091003  0.072575  0.077469  0.081787  0.000000  0.000000  0.695009
5  0.071872  0.078372  0.084491  0.107211  0.000000  0.000000  0.936480
6  0.078417  0.085083  0.112178  0.123985  0.000000  0.836902  0.708029
7  0.088283  0.112454  0.126067  0.150672  0.000000  0.000000  0.000000
8  0.112748  0.126124  0.152483  0.173243  0.201069  0.163153  0.273017
9  0.124711  0.149063  0.171720  0.201610  0.232600  0.270387  0.317934

          7         8         9
0  0.240278  0.271622  0.308231
1  0.267807  0.305760  0.353535
2  0.000000  0.000000  0.408816
3  0.604426  0.532933  0.469130
4  0.692911  0.605307  0.532406
5  0.801081  0.687060  0.595606
6  0.687057  0.599929  0.527933
7  0.492791  0.517581  0.464348
8  0.421480  0.447258  0.407768
9  0.368984  0.389756  0.360930
___policy___
       0      1      2      3      4      5      6      7      8     9
0  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT  DOWN
1  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  RIGHT  RIGHT  DOWN
2     UP   NONE   NONE   NONE   NONE   DOWN   NONE   NONE   NONE  DOWN
3     UP   LEFT   LEFT   DOWN   NONE  RIGHT   DOWN   DOWN   DOWN  LEFT
4     UP   LEFT   DOWN   DOWN   NONE   NONE   DOWN   DOWN   LEFT  LEFT
5     UP   DOWN   DOWN   DOWN   NONE   NONE   LEFT   LEFT   LEFT  LEFT
6   DOWN   DOWN   DOWN   DOWN   NONE     UP     UP     UP     UP  LEFT
7   DOWN   DOWN   DOWN   DOWN   NONE   NONE   NONE     UP     UP    UP
8  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT     UP     UP    UP
9  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT     UP     UP    UP

_________ iterations=100 discount=0.9 _________
average_square_error (measure of convergence) = 4.075041072630495e-07
___values___
          0         1         2         3         4         5         6  \
0  0.143403  0.162258  0.184807  0.210376  0.239231  0.271702  0.242390
1  0.153249  0.176482  0.203498  0.234844  0.271252  0.313611  0.273402
2  0.133580  0.000000  0.000000  0.000000  0.000000  0.366073  0.000000
3  0.115397  0.099927  0.087881  0.086882  0.000000  0.420170  0.599581
4  0.099947  0.088775  0.090083  0.098707  0.000000  0.000000  0.695005
5  0.088216  0.090596  0.101883  0.113978  0.000000  0.000000  0.936481
6  0.091695  0.102316  0.116916  0.131642  0.000000  0.836902  0.708025
7  0.103649  0.117143  0.133980  0.152205  0.000000  0.000000  0.000000
8  0.117871  0.133969  0.153590  0.175987  0.201049  0.163544  0.272873
9  0.131846  0.151002  0.174423  0.201698  0.233351  0.270186  0.318169

          7         8         9
0  0.240831  0.271576  0.308430
1  0.267755  0.306006  0.353445
2  0.000000  0.000000  0.408890
3  0.604405  0.532973  0.469072
4  0.692925  0.605279  0.532447
5  0.801075  0.687078  0.595576
6  0.687074  0.599891  0.527985
7  0.492754  0.517653  0.464266
```

```
8  0.421574  0.447125  0.407882
9  0.368856  0.389936  0.360787
___policy___
       0      1      2      3      4      5      6      7      8     9
0  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT  DOWN
1  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  RIGHT  RIGHT  DOWN
2     UP   NONE   NONE   NONE   NONE   DOWN   NONE   NONE   NONE  DOWN
3     UP   LEFT   LEFT   DOWN   NONE  RIGHT   DOWN   DOWN   DOWN  LEFT
4     UP   LEFT   DOWN   DOWN   NONE   NONE   DOWN   DOWN   LEFT  LEFT
5     UP   DOWN   DOWN   DOWN   NONE   NONE   LEFT   LEFT   LEFT  LEFT
6   DOWN   DOWN   DOWN   DOWN   NONE     UP     UP     UP     UP  LEFT
7   DOWN  RIGHT   DOWN   DOWN   NONE   NONE   NONE     UP     UP    UP
8  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT     UP     UP    UP
9  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT     UP     UP    UP

________ iterations=100 discount=0.9 ________
average_square_error (measure of convergence) = 4.6717626384234966e-11
___values___
          0         1         2         3         4         5         6  \
0  0.143408  0.162262  0.184807  0.210377  0.239231  0.271703  0.242390
1  0.153256  0.176484  0.203499  0.234844  0.271253  0.313611  0.273403
2  0.133590  0.000000  0.000000  0.000000  0.000000  0.366073  0.000000
3  0.115430  0.099987  0.088038  0.087084  0.000000  0.420170  0.599581
4  0.100006  0.088927  0.090178  0.098919  0.000000  0.000000  0.695005
5  0.088365  0.090684  0.102098  0.114031  0.000000  0.000000  0.936481
6  0.091812  0.102519  0.116938  0.131705  0.000000  0.836902  0.708025
7  0.103808  0.117165  0.134044  0.152211  0.000000  0.000000  0.000000
8  0.117901  0.134031  0.153592  0.176001  0.201048  0.163545  0.272873
9  0.131896  0.151010  0.174437  0.201697  0.233353  0.270185  0.318169

          7         8         9
0  0.240832  0.271576  0.308431
1  0.267754  0.306007  0.353445
2  0.000000  0.000000  0.408891
3  0.604405  0.532973  0.469072
4  0.692925  0.605279  0.532447
5  0.801075  0.687078  0.595576
6  0.687074  0.599891  0.527985
7  0.492754  0.517653  0.464266
8  0.421574  0.447125  0.407883
9  0.368855  0.389936  0.360787
___policy___
       0      1      2      3      4      5      6      7      8     9
0  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT  DOWN
1  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  RIGHT  RIGHT  DOWN
2     UP   NONE   NONE   NONE   NONE   DOWN   NONE   NONE   NONE  DOWN
3     UP   LEFT   LEFT   DOWN   NONE  RIGHT   DOWN   DOWN   DOWN  LEFT
4     UP   LEFT   DOWN   DOWN   NONE   NONE   DOWN   DOWN   LEFT  LEFT
5     UP   DOWN   DOWN   DOWN   NONE   NONE   LEFT   LEFT   LEFT  LEFT
6   DOWN   DOWN   DOWN   DOWN   NONE     UP     UP     UP     UP  LEFT
7   DOWN  RIGHT   DOWN   DOWN   NONE   NONE   NONE     UP     UP    UP
8  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT     UP     UP    UP
9  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT     UP     UP    UP

________ iterations=100 discount=0.9 ________
average_square_error (measure of convergence) = 1.5546103591866132e-15
```

```
___values___
          0         1         2         3         4         5         6  \
0  0.143408  0.162262  0.184807  0.210377  0.239231  0.271703  0.242390
1  0.153256  0.176484  0.203499  0.234844  0.271253  0.313611  0.273403
2  0.133590  0.000000  0.000000  0.000000  0.000000  0.366073  0.000000
3  0.115430  0.099987  0.088039  0.087085  0.000000  0.420170  0.599581
4  0.100006  0.088928  0.090179  0.098920  0.000000  0.000000  0.695005
5  0.088366  0.090684  0.102099  0.114031  0.000000  0.000000  0.936481
6  0.091812  0.102520  0.116938  0.131705  0.000000  0.836902  0.708025
7  0.103809  0.117165  0.134045  0.152211  0.000000  0.000000  0.000000
8  0.117901  0.134031  0.153592  0.176001  0.201048  0.163545  0.272873
9  0.131896  0.151010  0.174437  0.201697  0.233353  0.270185  0.318169

          7         8         9
0  0.240832  0.271576  0.308431
1  0.267754  0.306007  0.353445
2  0.000000  0.000000  0.408891
3  0.604405  0.532973  0.469072
4  0.692925  0.605279  0.532447
5  0.801075  0.687078  0.595576
6  0.687074  0.599891  0.527985
7  0.492754  0.517653  0.464266
8  0.421574  0.447125  0.407883
9  0.368855  0.389936  0.360787
___policy___
       0      1      2      3      4      5      6      7      8     9
0  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT  DOWN
1  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  RIGHT  RIGHT  DOWN
2     UP   NONE   NONE   NONE   NONE   DOWN   NONE   NONE   NONE  DOWN
3     UP   LEFT   LEFT   DOWN   NONE  RIGHT   DOWN   DOWN   DOWN  LEFT
4     UP   LEFT   DOWN   DOWN   NONE   NONE   DOWN   DOWN   LEFT  LEFT
5     UP   DOWN   DOWN   DOWN   NONE   NONE   LEFT   LEFT   LEFT  LEFT
6   DOWN   DOWN   DOWN   DOWN   NONE     UP     UP     UP     UP  LEFT
7   DOWN  RIGHT   DOWN   DOWN   NONE   NONE   NONE     UP     UP    UP
8  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT     UP     UP    UP
9  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT     UP     UP    UP

________ iterations=100 discount=0.9 ________
average_square_error (measure of convergence) = 3.3668587487262147e-20
___values___
          0         1         2         3         4         5         6  \
0  0.143408  0.162262  0.184807  0.210377  0.239231  0.271703  0.242390
1  0.153256  0.176484  0.203499  0.234844  0.271253  0.313611  0.273403
2  0.133590  0.000000  0.000000  0.000000  0.000000  0.366073  0.000000
3  0.115430  0.099987  0.088039  0.087085  0.000000  0.420170  0.599581
4  0.100006  0.088928  0.090179  0.098920  0.000000  0.000000  0.695005
5  0.088366  0.090684  0.102099  0.114031  0.000000  0.000000  0.936481
6  0.091812  0.102520  0.116938  0.131705  0.000000  0.836902  0.708025
7  0.103809  0.117165  0.134045  0.152211  0.000000  0.000000  0.000000
8  0.117901  0.134031  0.153592  0.176001  0.201048  0.163545  0.272873
9  0.131896  0.151010  0.174437  0.201697  0.233353  0.270185  0.318169

          7         8         9
0  0.240832  0.271576  0.308431
1  0.267754  0.306007  0.353445
2  0.000000  0.000000  0.408891
```

```
3  0.604405  0.532973  0.469072
4  0.692925  0.605279  0.532447
5  0.801075  0.687078  0.595576
6  0.687074  0.599891  0.527985
7  0.492754  0.517653  0.464266
8  0.421574  0.447125  0.407883
9  0.368855  0.389936  0.360787
___policy___
       0      1      2      3      4      5      6      7      8     9
0  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT  DOWN
1  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  RIGHT  RIGHT  DOWN
2     UP   NONE   NONE   NONE   NONE   DOWN   NONE   NONE   NONE  DOWN
3     UP   LEFT   LEFT   DOWN   NONE  RIGHT   DOWN   DOWN   DOWN  LEFT
4     UP   LEFT   DOWN   DOWN   NONE   NONE   DOWN   DOWN   LEFT  LEFT
5     UP   DOWN   DOWN   DOWN   NONE   NONE   LEFT   LEFT   LEFT  LEFT
6   DOWN   DOWN   DOWN   DOWN   NONE     UP     UP     UP     UP  LEFT
7   DOWN  RIGHT   DOWN   DOWN   NONE   NONE   NONE     UP     UP    UP
8  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT     UP     UP    UP
9  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT     UP     UP    UP

_________ iterations=100 discount=0.9 _________
average_square_error (measure of convergence) = 6.511526487921765e-25
___values___
         0         1         2         3         4         5         6  \
0  0.143408  0.162262  0.184807  0.210377  0.239231  0.271703  0.242390
1  0.153256  0.176484  0.203499  0.234844  0.271253  0.313611  0.273403
2  0.133590  0.000000  0.000000  0.000000  0.000000  0.366073  0.000000
3  0.115430  0.099987  0.088039  0.087085  0.000000  0.420170  0.599581
4  0.100006  0.088928  0.090179  0.098920  0.000000  0.000000  0.695005
5  0.088366  0.090684  0.102099  0.114031  0.000000  0.000000  0.936481
6  0.091812  0.102520  0.116938  0.131705  0.000000  0.836902  0.708025
7  0.103809  0.117165  0.134045  0.152211  0.000000  0.000000  0.000000
8  0.117901  0.134031  0.153592  0.176001  0.201048  0.163545  0.272873
9  0.131896  0.151010  0.174437  0.201697  0.233353  0.270185  0.318169

          7         8         9
0  0.240832  0.271576  0.308431
1  0.267754  0.306007  0.353445
2  0.000000  0.000000  0.408891
3  0.604405  0.532973  0.469072
4  0.692925  0.605279  0.532447
5  0.801075  0.687078  0.595576
6  0.687074  0.599891  0.527985
7  0.492754  0.517653  0.464266
8  0.421574  0.447125  0.407883
9  0.368855  0.389936  0.360787
___policy___
       0      1      2      3      4      5      6      7      8     9
0  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT  DOWN
1  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  RIGHT  RIGHT  DOWN
2     UP   NONE   NONE   NONE   NONE   DOWN   NONE   NONE   NONE  DOWN
3     UP   LEFT   LEFT   DOWN   NONE  RIGHT   DOWN   DOWN   DOWN  LEFT
4     UP   LEFT   DOWN   DOWN   NONE   NONE   DOWN   DOWN   LEFT  LEFT
5     UP   DOWN   DOWN   DOWN   NONE   NONE   LEFT   LEFT   LEFT  LEFT
6   DOWN   DOWN   DOWN   DOWN   NONE     UP     UP     UP     UP  LEFT
7   DOWN  RIGHT   DOWN   DOWN   NONE   NONE   NONE     UP     UP    UP
```

```
8  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT     UP     UP    UP
9  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT     UP     UP    UP

_________ iterations=100 discount=0.9 _________
average_square_error (measure of convergence) = 1.2975163690020128e-29
___values___
          0         1         2         3         4         5         6  \
0  0.143408  0.162262  0.184807  0.210377  0.239231  0.271703  0.242390
1  0.153256  0.176484  0.203499  0.234844  0.271253  0.313611  0.273403
2  0.133590  0.000000  0.000000  0.000000  0.000000  0.366073  0.000000
3  0.115430  0.099987  0.088039  0.087085  0.000000  0.420170  0.599581
4  0.100006  0.088928  0.090179  0.098920  0.000000  0.000000  0.695005
5  0.088366  0.090684  0.102099  0.114031  0.000000  0.000000  0.936481
6  0.091812  0.102520  0.116938  0.131705  0.000000  0.836902  0.708025
7  0.103809  0.117165  0.134045  0.152211  0.000000  0.000000  0.000000
8  0.117901  0.134031  0.153592  0.176001  0.201048  0.163545  0.272873
9  0.131896  0.151010  0.174437  0.201697  0.233353  0.270185  0.318169

          7         8         9
0  0.240832  0.271576  0.308431
1  0.267754  0.306007  0.353445
2  0.000000  0.000000  0.408891
3  0.604405  0.532973  0.469072
4  0.692925  0.605279  0.532447
5  0.801075  0.687078  0.595576
6  0.687074  0.599891  0.527985
7  0.492754  0.517653  0.464266
8  0.421574  0.447125  0.407883
9  0.368855  0.389936  0.360787
___policy___
       0      1      2      3      4      5      6      7      8     9
0  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT  DOWN
1  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  RIGHT  RIGHT  DOWN
2     UP   NONE   NONE   NONE   NONE   DOWN   NONE   NONE   NONE  DOWN
3     UP   LEFT   LEFT   DOWN   NONE  RIGHT   DOWN   DOWN   DOWN  LEFT
4     UP   LEFT   DOWN   DOWN   NONE   NONE   DOWN   DOWN   LEFT  LEFT
5     UP   DOWN   DOWN   DOWN   NONE   NONE   LEFT   LEFT   LEFT  LEFT
6   DOWN   DOWN   DOWN   DOWN   NONE     UP     UP     UP     UP  LEFT
7   DOWN  RIGHT   DOWN   DOWN   NONE   NONE   NONE     UP     UP    UP
8  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT     UP     UP    UP
9  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT     UP     UP    UP

_________ iterations=100 discount=0.9 _________
average_square_error (measure of convergence) = 2.996746993466539e-34
___values___
          0         1         2         3         4         5         6  \
0  0.143408  0.162262  0.184807  0.210377  0.239231  0.271703  0.242390
1  0.153256  0.176484  0.203499  0.234844  0.271253  0.313611  0.273403
2  0.133590  0.000000  0.000000  0.000000  0.000000  0.366073  0.000000
3  0.115430  0.099987  0.088039  0.087085  0.000000  0.420170  0.599581
4  0.100006  0.088928  0.090179  0.098920  0.000000  0.000000  0.695005
5  0.088366  0.090684  0.102099  0.114031  0.000000  0.000000  0.936481
6  0.091812  0.102520  0.116938  0.131705  0.000000  0.836902  0.708025
7  0.103809  0.117165  0.134045  0.152211  0.000000  0.000000  0.000000
8  0.117901  0.134031  0.153592  0.176001  0.201048  0.163545  0.272873
9  0.131896  0.151010  0.174437  0.201697  0.233353  0.270185  0.318169
```

```
          7         8         9
0  0.240832  0.271576  0.308431
1  0.267754  0.306007  0.353445
2  0.000000  0.000000  0.408891
3  0.604405  0.532973  0.469072
4  0.692925  0.605279  0.532447
5  0.801075  0.687078  0.595576
6  0.687074  0.599891  0.527985
7  0.492754  0.517653  0.464266
8  0.421574  0.447125  0.407883
9  0.368855  0.389936  0.360787
___policy___
       0      1      2      3      4      5      6      7      8     9
0  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT  DOWN
1  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  RIGHT  RIGHT  DOWN
2     UP   NONE   NONE   NONE   NONE   DOWN   NONE   NONE   NONE  DOWN
3     UP   LEFT   LEFT   DOWN   NONE  RIGHT   DOWN   DOWN   DOWN  LEFT
4     UP   LEFT   DOWN   DOWN   NONE   NONE   DOWN   DOWN   LEFT  LEFT
5     UP   DOWN   DOWN   DOWN   NONE   NONE   LEFT   LEFT   LEFT  LEFT
6   DOWN   DOWN   DOWN   DOWN   NONE     UP     UP     UP     UP  LEFT
7   DOWN  RIGHT   DOWN   DOWN   NONE   NONE   NONE     UP     UP    UP
8  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT     UP     UP    UP
9  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT     UP     UP    UP

________ iterations=100 discount=0.9 ________
average_square_error (measure of convergence) = 0.0
___values___
          0         1         2         3         4         5         6  \
0  0.143408  0.162262  0.184807  0.210377  0.239231  0.271703  0.242390
1  0.153256  0.176484  0.203499  0.234844  0.271253  0.313611  0.273403
2  0.133590  0.000000  0.000000  0.000000  0.000000  0.366073  0.000000
3  0.115430  0.099987  0.088039  0.087085  0.000000  0.420170  0.599581
4  0.100006  0.088928  0.090179  0.098920  0.000000  0.000000  0.695005
5  0.088366  0.090684  0.102099  0.114031  0.000000  0.000000  0.936481
6  0.091812  0.102520  0.116938  0.131705  0.000000  0.836902  0.708025
7  0.103809  0.117165  0.134045  0.152211  0.000000  0.000000  0.000000
8  0.117901  0.134031  0.153592  0.176001  0.201048  0.163545  0.272873
9  0.131896  0.151010  0.174437  0.201697  0.233353  0.270185  0.318169

          7         8         9
0  0.240832  0.271576  0.308431
1  0.267754  0.306007  0.353445
2  0.000000  0.000000  0.408891
3  0.604405  0.532973  0.469072
4  0.692925  0.605279  0.532447
5  0.801075  0.687078  0.595576
6  0.687074  0.599891  0.527985
7  0.492754  0.517653  0.464266
8  0.421574  0.447125  0.407883
9  0.368855  0.389936  0.360787
___policy___
       0      1      2      3      4      5      6      7      8     9
0  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT  DOWN
1  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT   DOWN   LEFT  RIGHT  RIGHT  DOWN
2     UP   NONE   NONE   NONE   NONE   DOWN   NONE   NONE   NONE  DOWN
```

```
3     UP   LEFT   LEFT   DOWN   NONE  RIGHT   DOWN   DOWN   DOWN  LEFT
4     UP   LEFT   DOWN   DOWN   NONE   NONE   DOWN   DOWN   LEFT  LEFT
5     UP   DOWN   DOWN   DOWN   NONE   NONE   LEFT   LEFT   LEFT  LEFT
6   DOWN   DOWN   DOWN   DOWN   NONE     UP     UP     UP     UP  LEFT
7   DOWN  RIGHT   DOWN   DOWN   NONE   NONE   NONE     UP     UP    UP
8  RIGHT  RIGHT  RIGHT   DOWN   DOWN  RIGHT  RIGHT     UP     UP    UP
9  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT  RIGHT     UP     UP    UP
```

1. ¿Cuántas muestras debe tomar el agente? Su implementación no debe utilizar este número como un parámetro o tenerlo como un factor predeterminado del agente.
    - En el caso del metodo de monte carlo el agente converge a una estrategia en alrededor de 1000 iteraciones; sin embargo, hay ciertas estrategias que no consigue (más adelante en el archivo se explica por qué)
2. ¿Cómo se comparan los valores de `MCM` con respecto a los valores obtenidos en el ejercicio de iteración de valores `value_iteration`? ¿Por qué se da la diferencia si existe alguna, o por qué no existe ninguna diferencia? La diferencia es que el agente MCM no encuentra la estrategia pasando por el estrecho en las coordenadas (5, 3) esto es debido a que la estrategia es muy sensible a errores, por lo tanto, si el epsilon es alto la estrategia no será efectiva en las simulaciones. Por otro lado, si el epsilon es muy bajo no se encontrará la estrategia porque el agente no explorar ese espacio.
3. ¿Cómo se compara la política obtenida utilizando `MCM` con respecto a la política obtenida en el ejercicio de iteración de políticas `policy_iteration`? ¿Por qué se da la diferencia si existe alguna, o por qué no existe ninguna diferencia? Los resultados difieren en la misma forma que lo hace la iteration de valores por las mismas razones
4. ¿Cuál es el efecto de del factor de descuento sobre el método de Monte Carlo, calcule la solución de Gridworld con diferentes valores? En el caso de un descuento de uno el agente ya no busca los caminos más cortos, sino los más seguros, es decir optimiza evitar los -1s más que acercarse. en el caso del descuento muy bajo el agente se arriesga más para llegar más rapido al objetivo