



```
In [2]: # Universidad Politécnica Salesiana - UPS
# Estudiante: Rafael Angamarca
# Examen Inteligencia Artificial
# Desarrollo
```

```
In [3]: #Probar las libreria y hacer un pequeño
#Gym es un conjunto de herramientas para
#desarrollar y comparar algoritmos de aprendizaje por refuerzo.
# No hace suposiciones sobre la estructura de su agente y es compatible
# con cualquier biblioteca de cálculo numérico, como TensorFlow o Theano.
```

```
In [4]: #Qué es openAI gym
#nos da un gran número de entornos de prueba para trabajar
#en los algoritmos de nuestro agente RL con interfaces compartidas
```

```
pip install gym
```

```
Requirement already satisfied: gym in c:\programdata\anaconda3\lib\site-packages (0.18.3)
Requirement already satisfied: Pillow<=8.2.0 in c:\programdata\anaconda3\lib\site-packages (from gym) (8.2.0)
Requirement already satisfied: numpy>=1.10.4 in c:\programdata\anaconda3\lib\site-packages (from gym) (1.20.1)
Requirement already satisfied: scipy in c:\programdata\anaconda3\lib\site-packages (from gym) (1.6.2)
Requirement already satisfied: cloudpickle<1.7.0,>=1.2.0 in c:\programdata\anaconda3\lib\site-packages (from gym) (1.6.0)
Requirement already satisfied: pygame<=1.5.15,>=1.4.0 in c:\programdata\anaconda3\lib\site-packages (from gym) (1.5.15)
Note: you may need to restart the kernel to use updated packages.
```

```
In [5]: #Puedes ver la Lista de entornos disponibles en tu instalación de Gym de la siguiente manera:
```

```
from gym import envs
print(envs.registry.all())
```

```
dict_values([EnvSpec(Copy-v0), EnvSpec(RepeatCopy-v0), EnvSpec(ReversedAddition-v0), EnvSpec(ReversedAddition3-v0), EnvSpec(DuplicatedInput-v0), EnvSpec(Reverse-v0), EnvSpec(CartPole-v0), EnvSpec(CartPole-v1), EnvSpec(MountainCar-v0), EnvSpec(MountainCarContinuous-v0), EnvSpec(Pendulum-v0), EnvSpec(Acrobot-v1), EnvSpec(LunarLander-v2), EnvSpec(LunarLanderContinuous-v2), EnvSpec(BipedalWalker-v3), EnvSpec(BipedalWalkerHardcore-v3), EnvSpec(CarRacing-v0), EnvSpec(Blackjack-v0), EnvSpec(KellyCoinflip-v0), EnvSpec(KellyCoinflipGeneralized-v0), EnvSpec(FrozenLake-v0), EnvSpec(FrozenLake8x8-v0), EnvSpec(CliffWalking-v0), EnvSpec(NChain-v0), EnvSpec(Roulette-v0), EnvSpec(Taxi-v3), EnvSpec(GuessingGame-v0), EnvSpec(HotterColder-v0), EnvSpec(Reacher-v2), EnvSpec(Pusher-v2), EnvSpec(Thrower-v2), EnvSpec(Striker-v2), EnvSpec(InvertedPendulum-v2), EnvSpec(InvertedDoublePendulum-v2), EnvSpec(HalfCheetah-v2), EnvSpec(HalfCheetah-v3), EnvSpec(Hopper-v2), EnvSpec(Hopper-v3), EnvSpec(Swimmer-v2), EnvSpec(Swimmer-v3), EnvSpec(Walker2d-v2), EnvSpec(Walker2d-v3), EnvSpec(Ant-v2), EnvSpec(Ant-v3), EnvSpec(Humanoid-v2), EnvSpec(Humanoid-v3), EnvSpec(Humanoid-v4)])
```

```
FHHFFHF
FHHFFHF
FFFHFFG
(Right)
SFFFFFF
FFFFFF
FFFFFF
FFFFFF
FFFFFF
FHHFFHF
FHHFFHF
FFFHFFG
(Up)
FFFFFF
```

In []:

```
In [2]: # Reset environment
s = env.reset()
d = False
# The Q-Table Learning algorithm
while d != True:
    env.render()
    # Choose action from Q table
    a = np.argmax(Q[s,:] + np.random.randn(1,env.action_space.n)*(1./(i+1)))
    #Get new state & reward from environment
    s1,r,d,_ = env.step(a)
    #Update Q-Table with new knowledge
    Q[s,a] = Q[s,a] + eta*(r + gma*np.max(Q[s1,:]) - Q[s,a])
    s = s1
# Code will stop at d == True, and render one state before it
```

```
FFFFFF
FFFFFF
FFFHFF
FFFFHF
FFFFHF
FHHFFHF
FHHFFHF
FFFHFFG
(Up)
FFFFFF
```

FFFFHFFF
FHHFFFFH
FHFFHFFH
FFFHFFFG
(Right)
SFFFFFFF
FFHFFFFF
FFFHFFFF
FFFFFHFF
FFFHFFFF
FHHFFFFH
FHFFHFFH
FFFHFFFG
(Up)
SFHFFFFF
FFFFFHFF
FFFFHFFF

```
#Conclusiones
#Gym también es compatible con TensorFlow & PyTorch.
# En la actualidad la IA va tomando fuerza en varios campos
```

```
#Segunda Parte de Examen.
#Dentro del juego el usuario deberá escoger/ingresar su ciudad natal incluido latitud y longitud#
#En base a ello recomendar usuarios cercanos utilizando el algoritmos A* y Yenn,
#se debe tener una base de datos de al menos 50 usuarios dentro de una misma ciudad
#(Tomar datos de pruebas anteriores o generar una nueva base de datos),
# tener en presente que el árbol debe tener al menos 7 niveles o superior y con 3 conexiones de nodos cada uno.
```