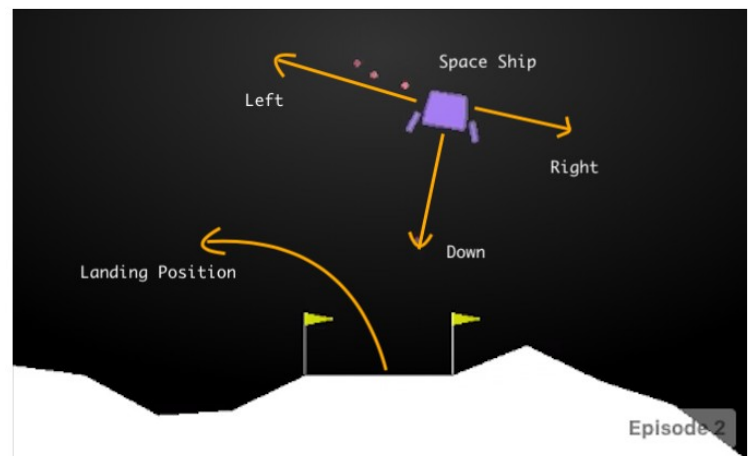
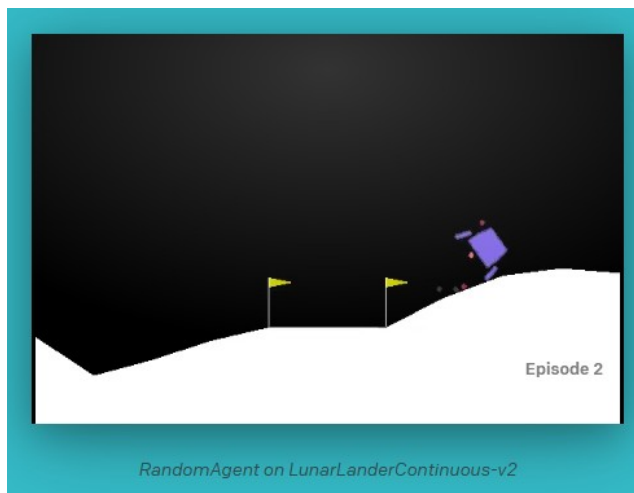


## Exercise: Deep Reinforcement Learning

Write a program that solves the Lunar Lander Problem as described by OpenAI Gym using DeepRL methods. Note, that the problem is stated in discrete action (do nothing, fire left orientation engine, fire main engine, fire right orientation engine) and discrete environment states (coordinates of the lander in x & y, its linear velocities in x & y, its angle, its angular velocity, and two booleans that represent whether each leg is in contact with the ground or not).

“Rocket trajectory optimization is a classic topic in Optimal Control.

According to Pontryagin's maximum principle it's optimal to fire engine full throttle or turn it off. That's the reason this environment is OK to have discrete actions (engine on or off).



The landing pad is always at coordinates (0,0). The coordinates are the first two numbers in the state vector. Reward for moving from the top of the screen to the landing pad and zero speed is about 100..140 points. If the lander moves away from the landing pad it loses reward. The episode finishes if the lander crashes or comes to rest, receiving an additional -100 or +100 points. Each leg with ground contact is +10 points. Firing the main engine is -0.3 points each frame. Firing the side engine is -0.03 points each frame. Solved is 200 points.”

[https://www.gymnasium.ml/environments/box2d/lunar\\_lander/](https://www.gymnasium.ml/environments/box2d/lunar_lander/)

<https://towardsdatascience.com/reinforcement-learning-with-openai-d445c2c687d2>

[https://github.com/openai/gym/blob/master/gym/envs/box2d/lunar\\_lander.py](https://github.com/openai/gym/blob/master/gym/envs/box2d/lunar_lander.py)

**Note** (you might also need to install box2d & pygame):

> pip install gym

> pip install box2d

> pip install pygame