Short summary of the exercise topic

The third assignment is the BipedalWalker-v3 challenge, a complex reinforcement learning task involving two-legged agents learning how to walk without falling. The task involves developing a model capable of learning to balance and walk through trial and error, using a combination of policy (Actor) and value (Critic) network updates.

How you solved the problem

The solution for the assignment can be split into different blocks.

For the class Agent:

- Action and Log Probability Calculation: This method prepares state tensors, checks their shape, computes actions from the current policy (via sampling), and calculates the corresponding log probabilities.
- Training Loop: Processes batches from the buffer, calculating advantages, target values, and policy updates using the clipped surrogate objective from Proximal Policy Optimization (PPO). It includes loss computation for policy, value, and entropy to ensure exploration.

For the class Actor:

- Network Architecture: Composed of fully connected layers with normalization to stabilize the inputs. Outputs action mean and sigma for continuous action spaces.
- Forward Pass Implementation: Calculates action distributions using RELU activations for non-linearity and stability in the action outputs.

For the class Critic:

- Value Estimation: Similar architecture to the Actor, but concludes with a single output that estimates the value of the current state, helping in advantage computation.
- Forward Functionality: Processes input states through fully connected layers to estimate the state's value, which aids in determining the policy's performance.

For the class Runner:

- Environment Interaction Loop: Manages episodes, collecting data through interactions with the environment using the policies defined by the agent.
- Buffer Management and Statistics: Adds episodes to the buffer, resets buffers when necessary, and logs statistics like rewards and episode lengths for performance tracking and model evaluation.

Hyperparameters Selection:

- After different tries, the used hyperparameters are written in the code.

Performance of your agent in your own evaluation (>280 in server)

The agent demonstrated substantial improvements in walking capability over training episodes. Initially, the agent struggled with maintaining balance and moving forward effectively. However, as training progressed, it began to adapt, learning to stabilize and accelerate its pace across the terrain.