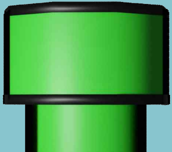
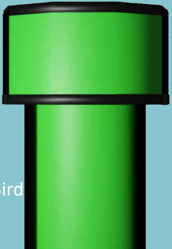
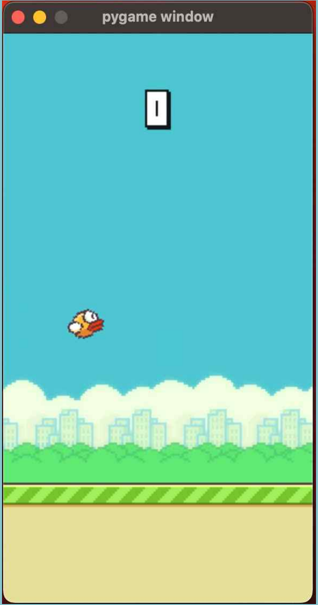
# FLAPPY BIRD PROJECT

김정찬, 홍유빈

https://github.com/hwruchan/FlappyBird



# Flappy bird

터치 또는 클릭 하면 FLY
아무 것도 하지 않으면 FALL
파이프를 통과하며 멀리 오래 살아남는 게임

# PROJECT AIM

**1. Learning amount :**

Episode 수(10,000개 100,000개 500,000개)에 따라 학습 능률의 결과 비교

**2. Hidden layer :**

Hidden layer의 개수(1개, 2개)에 따른 비교

**3. Hyperparameter :**

Hyperparameter 값 조정(discount factor)의 따른 비교

**4. Naïve DQN :**

Naïve DQN 과 DQN 의 성능 비교

**5. Custom Environment**

# STATE

# Gymnasium

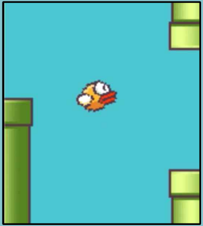
| | | |
|---|---|---|
| the last pipe's horizontal position | -> | 마지막으로 나타난 파이프의 x 좌표. |
| the last top pipe's vertical position | -> | 마지막으로 나타난 상단 파이프의 y 좌표. |
| the last bottom pipe's vertical position | -> | 마지막으로 나타난 하단 파이프의 y 좌표. |
| the next pipe's horizontal position | -> | 다음에 나타날 파이프의 x 좌표. |
| the next top pipe's vertical position | -> | 다음에 나타날 상단 파이프의 y 좌표. |
| the next bottom pipe's vertical position | -> | 다음에 나타날 하단 파이프의 y 좌표. |
| the next next pipe's horizontal position | -> | 그 다음에 나타날 파이프의 x 좌표. |
| the next next top pipe's vertical position | -> | 그 다음에 나타날 상단 파이프의 y 좌표. |
| the next next bottom pipe's vertical position | -> | 그 다음에 나타날 하단 파이프의 y 좌표. |
| player's vertical position | -> | 플레이어(새)의 y 좌표. |
| player's vertical velocity | -> | 플레이어의 y 방향 속도. |
| player's rotation | -> | 플레이어의 회전 각도. |

# ACTION

JUMP

Nothing

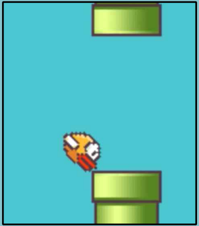
# Reward



<every frame it stays alive>

살아남은 프레임 당

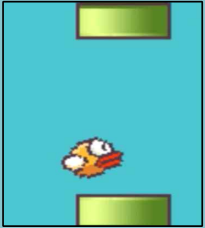
+ 0.1



<DYING>

죽었을 때

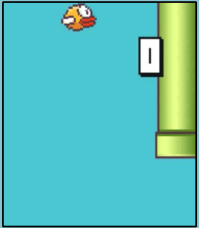
- 1.0



<successfully passing a pipe>

파이프 통과할 때

+ 1



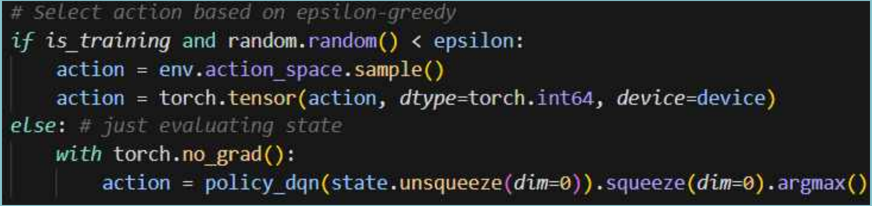
<touch the top of the screen>

화면 상단에 닿았을 때

- 0.5

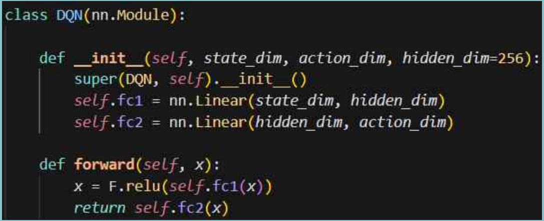
# CODE

# Epsilon greedy Policy

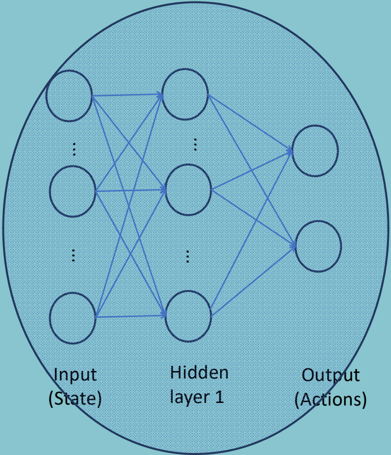


```
# Select action based on epsilon-greedy
if is_training and random.random() < epsilon:
    action = env.action_space.sample()
    action = torch.tensor(action, dtype=torch.int64, device=device)
else: # just evaluating state
    with torch.no_grad():
        action = policy_dqn(state.unsqueeze(dim=0)).squeeze(dim=0).argmax()
```

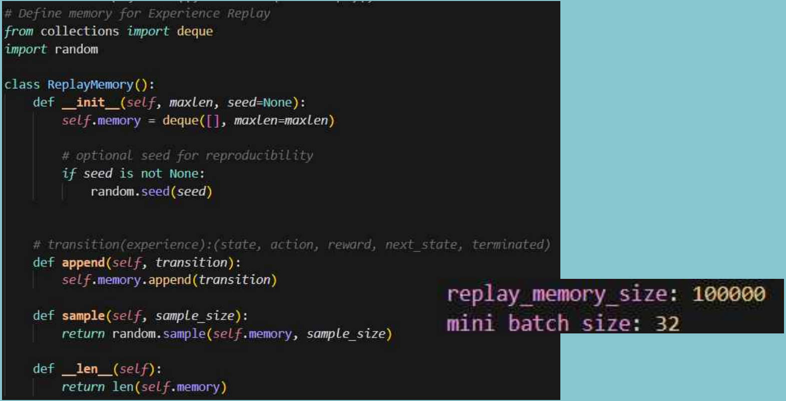
# DQN



```
class DQN(nn.Module):

    def __init__(self, state_dim, action_dim, hidden_dim=256):
        super(DQN, self).__init__()
        self.fc1 = nn.Linear(state_dim, hidden_dim)
        self.fc2 = nn.Linear(hidden_dim, action_dim)

    def forward(self, x):
        x = F.relu(self.fc1(x))
        return self.fc2(x)
```

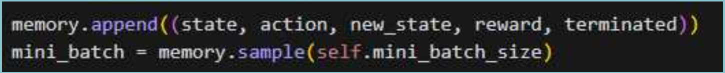


# Replay Buffer



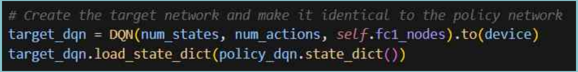
```python
# Define memory for Experience Replay
from collections import deque
import random

class ReplayMemory():
    def __init__(self, maxlen, seed=None):
        self.memory = deque([], maxlen=maxlen)

        # optional seed for reproducibility
        if seed is not None:
            random.seed(seed)


    # transition(experience):(state, action, reward, next_state, terminated)
    def append(self, transition):
        self.memory.append(transition)

    def sample(self, sample_size):
        return random.sample(self.memory, sample_size)

    def __len__(self):
        return len(self.memory)
```

```
replay_memory_size: 100000
mini batch size: 32
```

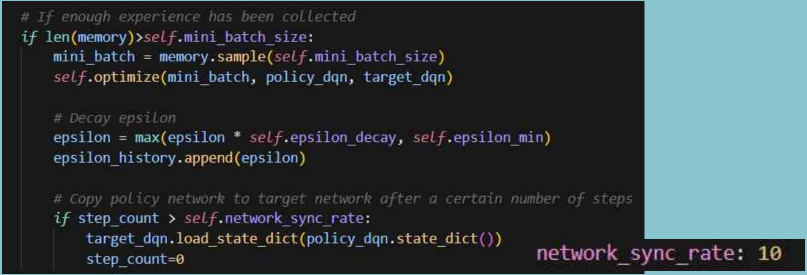


```python
memory.append((state, action, new_state, reward, terminated))
mini_batch = memory.sample(self.mini_batch_size)
```

# Target Network



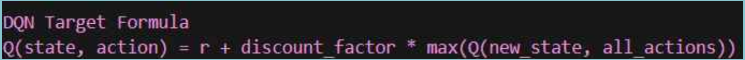
```
# Create the target network and make it identical to the policy network
target_dqn = DQN(num_states, num_actions, self.fc1_nodes).to(device)
target_dqn.load_state_dict(policy_dqn.state_dict())
```



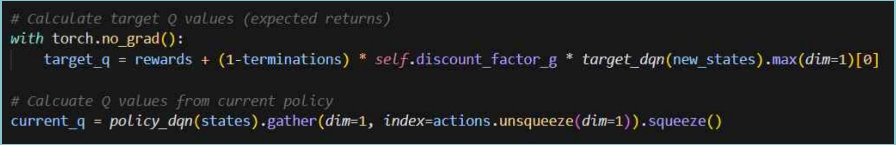
```
# If enough experience has been collected
if len(memory)>self.mini_batch_size:
    mini_batch = memory.sample(self.mini_batch_size)
    self.optimize(mini_batch, policy_dqn, target_dqn)

    # Decay epsilon
    epsilon = max(epsilon * self.epsilon_decay, self.epsilon_min)
    epsilon_history.append(epsilon)

    # Copy policy network to target network after a certain number of steps
    if step_count > self.network_sync_rate:
        target_dqn.load_state_dict(policy_dqn.state_dict())
        step_count=0
```

```
network_sync_rate: 10
```

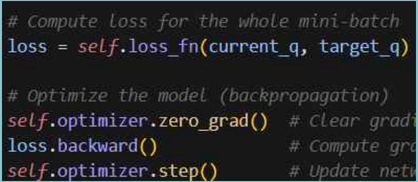
# Q



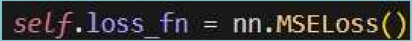
```
DQN Target Formula
Q(state, action) = r + discount_factor * max(Q(new_state, all_actions))
```



```python
# Calculate target Q values (expected returns)
with torch.no_grad():
    target_q = rewards + (1-terminations) * self.discount_factor_g * target_dqn(new_states).max(dim=1)[0]

# Calcuate Q values from current policy
current_q = policy_dqn(states).gather(dim=1, index=actions.unsqueeze(dim=1)).squeeze()
```



```python
# Compute loss for the whole mini-batch
loss = self.loss_fn(current_q, target_q)

# Optimize the model (backpropagation)
self.optimizer.zero_grad()  # Clear gradi
loss.backward()             # Compute gra
self.optimizer.step()       # Update netw
```



```python
self.loss_fn = nn.MSELoss()
```

# FIRST AIM

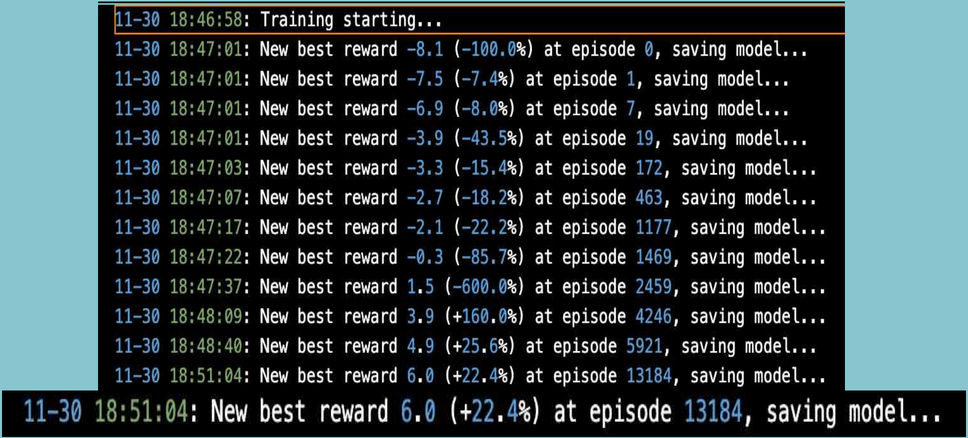
# Amount of Learning

10,000

100,000

500,000

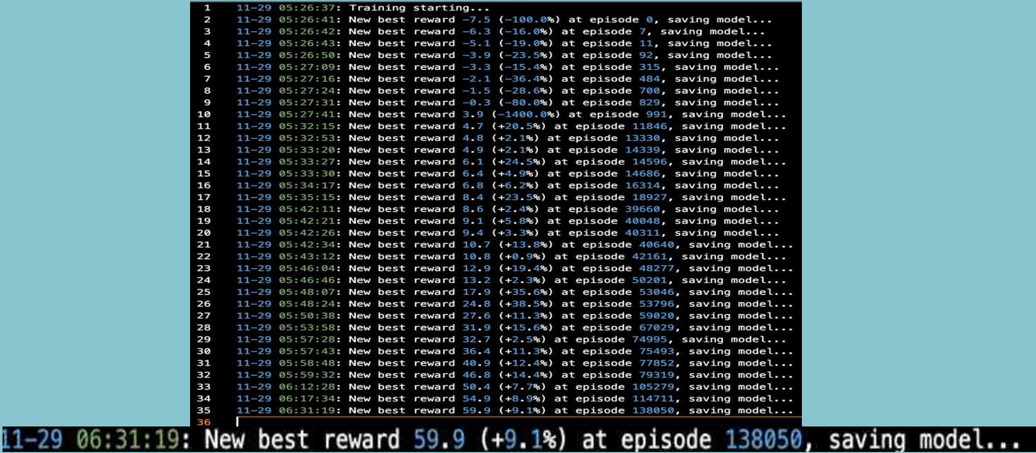
# 10,000 Episodes



```
11-30 18:46:58: Training starting...
11-30 18:47:01: New best reward -8.1 (-100.0%) at episode 0, saving model...
11-30 18:47:01: New best reward -7.5 (-7.4%) at episode 1, saving model...
11-30 18:47:01: New best reward -6.9 (-8.0%) at episode 7, saving model...
11-30 18:47:01: New best reward -3.9 (-43.5%) at episode 19, saving model...
11-30 18:47:03: New best reward -3.3 (-15.4%) at episode 172, saving model...
11-30 18:47:07: New best reward -2.7 (-18.2%) at episode 463, saving model...
11-30 18:47:17: New best reward -2.1 (-22.2%) at episode 1177, saving model...
11-30 18:47:22: New best reward -0.3 (-85.7%) at episode 1469, saving model...
11-30 18:47:37: New best reward 1.5 (-600.0%) at episode 2459, saving model...
11-30 18:48:09: New best reward 3.9 (+160.0%) at episode 4246, saving model...
11-30 18:48:40: New best reward 4.9 (+25.6%) at episode 5921, saving model...
11-30 18:51:04: New best reward 6.0 (+22.4%) at episode 13184, saving model...
```

```
11-30 18:51:04: New best reward 6.0 (+22.4%) at episode 13184, saving model...
```

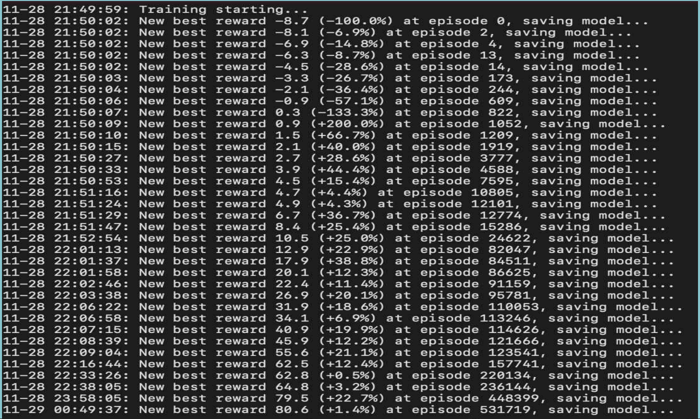
# 100,000 Episodes



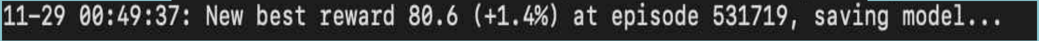
```
11-29 05:26:37: Training starting...
11-29 05:26:41: New best reward -7.5 (-100.0%) at episode 0, saving model...
11-29 05:26:42: New best reward -6.3 (-16.0%) at episode 7, saving model...
11-29 05:26:43: New best reward -5.1 (-19.0%) at episode 11, saving model...
11-29 05:26:50: New best reward -3.9 (-23.5%) at episode 92, saving model...
11-29 05:27:09: New best reward -3.3 (-15.4%) at episode 315, saving model...
11-29 05:27:16: New best reward -2.1 (-36.4%) at episode 484, saving model...
11-29 05:27:24: New best reward -1.5 (-28.6%) at episode 700, saving model...
11-29 05:27:31: New best reward -0.3 (-80.0%) at episode 829, saving model...
11-29 05:27:41: New best reward 3.9 (-1400.0%) at episode 991, saving model...
11-29 05:32:15: New best reward 4.7 (+20.5%) at episode 11846, saving model...
11-29 05:32:53: New best reward 4.8 (+2.1%) at episode 13330, saving model...
11-29 05:33:20: New best reward 4.9 (+2.1%) at episode 14339, saving model...
11-29 05:33:27: New best reward 6.1 (+24.5%) at episode 14596, saving model...
11-29 05:33:30: New best reward 6.4 (+4.9%) at episode 14686, saving model...
11-29 05:34:17: New best reward 6.8 (+6.2%) at episode 16314, saving model...
11-29 05:35:15: New best reward 8.4 (+23.5%) at episode 18927, saving model...
11-29 05:42:11: New best reward 8.6 (+2.4%) at episode 39660, saving model...
11-29 05:42:21: New best reward 9.1 (+5.8%) at episode 40048, saving model...
11-29 05:42:26: New best reward 9.4 (+3.3%) at episode 40311, saving model...
11-29 05:42:34: New best reward 10.7 (+13.8%) at episode 40640, saving model...
11-29 05:43:12: New best reward 10.8 (+0.9%) at episode 42161, saving model...
11-29 05:46:04: New best reward 12.9 (+19.4%) at episode 48277, saving model...
11-29 05:46:46: New best reward 13.2 (+2.3%) at episode 50201, saving model...
11-29 05:48:07: New best reward 17.9 (+35.6%) at episode 53046, saving model...
11-29 05:48:24: New best reward 24.8 (+38.5%) at episode 53796, saving model...
11-29 05:50:38: New best reward 27.6 (+11.3%) at episode 59020, saving model...
11-29 05:53:58: New best reward 31.9 (+15.6%) at episode 67029, saving model...
11-29 05:57:28: New best reward 32.7 (+2.5%) at episode 74995, saving model...
11-29 05:57:43: New best reward 36.4 (+11.3%) at episode 75493, saving model...
11-29 05:58:48: New best reward 40.9 (+12.4%) at episode 77852, saving model...
11-29 05:59:32: New best reward 46.8 (+14.4%) at episode 79319, saving model...
11-29 06:12:28: New best reward 50.4 (+7.7%) at episode 105279, saving model...
11-29 06:17:34: New best reward 54.9 (+8.9%) at episode 114711, saving model...
11-29 06:31:19: New best reward 59.9 (+9.1%) at episode 138050, saving model...
```

11-29 06:31:19: New best reward 59.9 (+9.1%) at episode 138050, saving model...

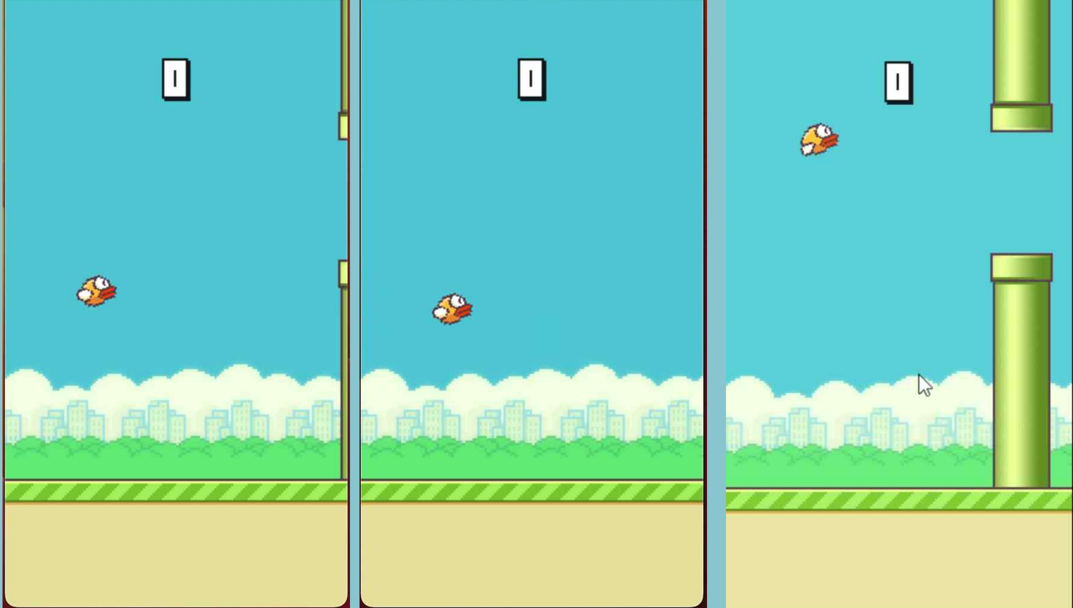
# 500,000 Episodes

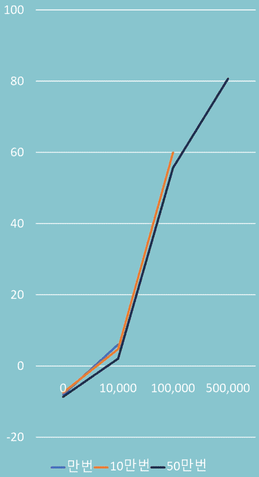


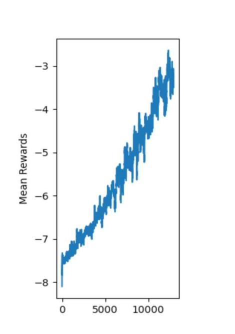
```
11-28 21:49:59: Training starting...
11-28 21:50:02: New best reward -8.7 (-100.0%) at episode 0, saving model...
11-28 21:50:02: New best reward -8.1 (-6.9%) at episode 2, saving model...
11-28 21:50:02: New best reward -6.9 (-14.8%) at episode 4, saving model...
11-28 21:50:02: New best reward -6.3 (-8.7%) at episode 13, saving model...
11-28 21:50:02: New best reward -4.5 (-28.6%) at episode 14, saving model...
11-28 21:50:03: New best reward -3.3 (-26.7%) at episode 173, saving model...
11-28 21:50:04: New best reward -2.1 (-36.4%) at episode 244, saving model...
11-28 21:50:06: New best reward -0.9 (-57.1%) at episode 609, saving model...
11-28 21:50:07: New best reward 0.3 (-133.3%) at episode 822, saving model...
11-28 21:50:09: New best reward 0.9 (+200.0%) at episode 1052, saving model...
11-28 21:50:10: New best reward 1.5 (+66.7%) at episode 1209, saving model...
11-28 21:50:15: New best reward 2.1 (+40.0%) at episode 1919, saving model...
11-28 21:50:27: New best reward 2.7 (+28.6%) at episode 3777, saving model...
11-28 21:50:33: New best reward 3.9 (+44.4%) at episode 4588, saving model...
11-28 21:50:53: New best reward 4.5 (+15.4%) at episode 7595, saving model...
11-28 21:51:16: New best reward 4.7 (+4.4%) at episode 10805, saving model...
11-28 21:51:24: New best reward 4.9 (+4.3%) at episode 12101, saving model...
11-28 21:51:29: New best reward 6.7 (+36.7%) at episode 12774, saving model...
11-28 21:51:47: New best reward 8.4 (+25.4%) at episode 15286, saving model...
11-28 21:52:54: New best reward 10.5 (+25.0%) at episode 24622, saving model...
11-28 22:01:13: New best reward 12.9 (+22.9%) at episode 82047, saving model...
11-28 22:01:37: New best reward 17.9 (+38.8%) at episode 84511, saving model...
11-28 22:01:58: New best reward 20.1 (+12.3%) at episode 86625, saving model...
11-28 22:02:46: New best reward 22.4 (+11.4%) at episode 91159, saving model...
11-28 22:03:38: New best reward 26.9 (+20.1%) at episode 95781, saving model...
11-28 22:06:22: New best reward 31.9 (+18.6%) at episode 110053, saving model...
11-28 22:06:58: New best reward 34.1 (+6.9%) at episode 113246, saving model...
11-28 22:07:15: New best reward 40.9 (+19.9%) at episode 114626, saving model...
11-28 22:08:39: New best reward 45.9 (+12.2%) at episode 121666, saving model...
11-28 22:09:04: New best reward 55.6 (+21.1%) at episode 123541, saving model...
11-28 22:16:44: New best reward 62.5 (+12.4%) at episode 157741, saving model...
11-28 22:33:26: New best reward 62.8 (+0.5%) at episode 220134, saving model...
11-28 22:38:05: New best reward 64.8 (+3.2%) at episode 236144, saving model...
11-28 23:58:05: New best reward 79.5 (+22.7%) at episode 448399, saving model...
11-29 00:49:37: New best reward 80.6 (+1.4%) at episode 531719, saving model...
```



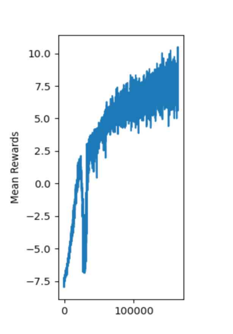
```
11-29 00:49:37: New best reward 80.6 (+1.4%) at episode 531719, saving model...
```


1
1
1

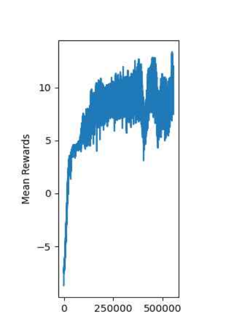




**10,000 episode**
**Max Reward: 6.0**
**Time taken: 5m**



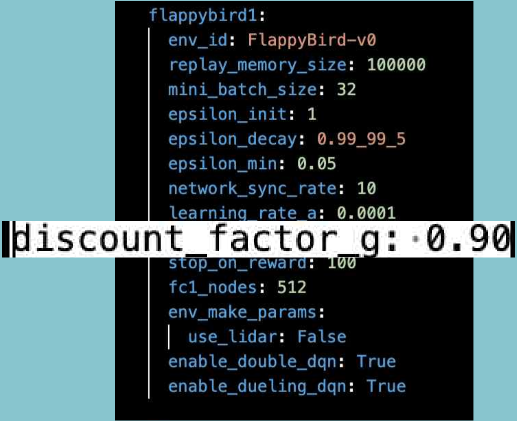
**100,000 episode**
**Max Reward : 50**
**Time taken: 46m**



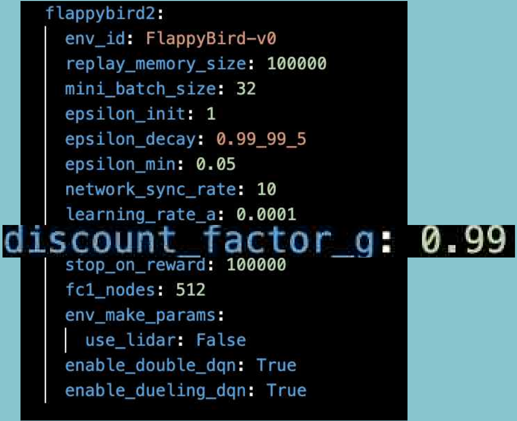
**500,000 episode**
**Max Reward : 80**
**Time taken: 3h**

# SECOND AIM

지금 당장의 파이프만 넘으면
되는데 미래의 보상이
중요한가...?



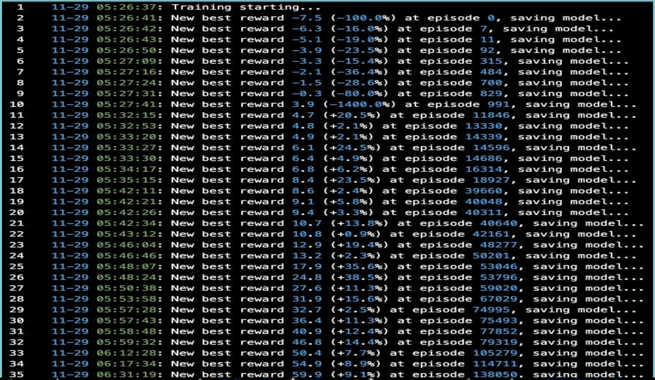
```
flappybird1:
  env_id: FlappyBird-v0
  replay_memory_size: 100000
  mini_batch_size: 32
  epsilon_init: 1
  epsilon_decay: 0.99_99_5
  epsilon_min: 0.05
  network_sync_rate: 10
  learning_rate_a: 0.0001
  discount_factor_g: 0.90
  stop_on_reward: 100
  fc1_nodes: 512
  env_make_params:
    use_lidar: False
  enable_double_dqn: True
  enable_dueling_dqn: True
```



```
flappybird2:
  env_id: FlappyBird-v0
  replay_memory_size: 100000
  mini_batch_size: 32
  epsilon_init: 1
  epsilon_decay: 0.99_99_5
  epsilon_min: 0.05
  network_sync_rate: 10
  learning_rate_a: 0.0001
  discount_factor_g: 0.99
  stop_on_reward: 100000
  fc1_nodes: 512
  env_make_params:
    use_lidar: False
  enable_double_dqn: True
  enable_dueling_dqn: True
```

즉각적인 행동의 결과에 더 집중해!

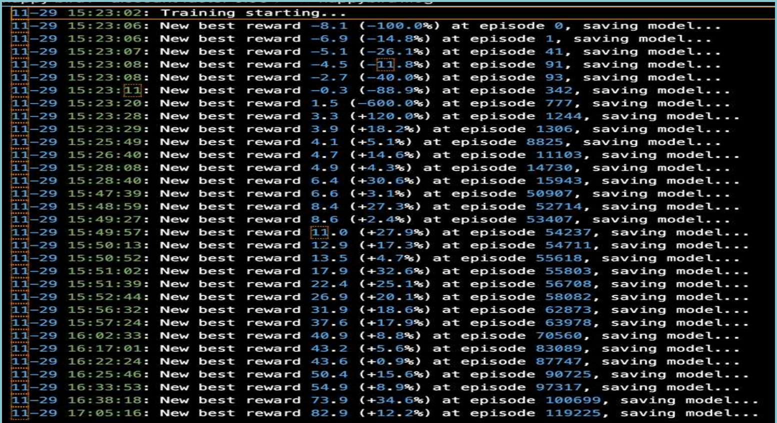
# Discount Factor : 0.99



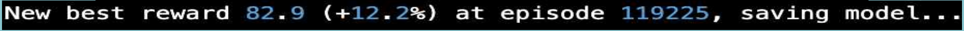
```
11-29 05:26:37: Training starting...
11-29 05:26:41: New best reward -7.5 (-100.0%) at episode 0, saving model...
11-29 05:26:42: New best reward -6.3 (-16.0%) at episode 7, saving model...
11-29 05:26:43: New best reward -5.1 (-19.0%) at episode 11, saving model...
11-29 05:26:50: New best reward -3.9 (-23.5%) at episode 92, saving model...
11-29 05:27:09: New best reward -3.3 (-15.4%) at episode 315, saving model...
11-29 05:27:16: New best reward -2.1 (-36.4%) at episode 484, saving model...
11-29 05:27:24: New best reward -1.5 (-28.6%) at episode 700, saving model...
11-29 05:27:31: New best reward -0.3 (-80.0%) at episode 829, saving model...
11-29 05:27:41: New best reward 3.9 (-1400.0%) at episode 991, saving model...
11-29 05:32:15: New best reward 4.7 (+20.5%) at episode 11846, saving model...
11-29 05:32:53: New best reward 4.8 (+2.1%) at episode 13330, saving model...
11-29 05:33:20: New best reward 4.9 (+2.1%) at episode 14339, saving model...
11-29 05:33:27: New best reward 6.1 (+24.5%) at episode 14596, saving model...
11-29 05:33:30: New best reward 6.4 (+4.9%) at episode 14686, saving model...
11-29 05:34:17: New best reward 6.8 (+6.2%) at episode 16314, saving model...
11-29 05:35:15: New best reward 8.4 (+23.5%) at episode 18927, saving model...
11-29 05:42:11: New best reward 8.6 (+2.4%) at episode 39660, saving model...
11-29 05:42:21: New best reward 9.1 (+5.8%) at episode 40048, saving model...
11-29 05:42:26: New best reward 9.4 (+3.3%) at episode 40311, saving model...
11-29 05:42:34: New best reward 10.7 (+13.8%) at episode 40640, saving model...
11-29 05:43:12: New best reward 10.8 (+0.9%) at episode 42161, saving model...
11-29 05:46:04: New best reward 12.9 (+19.4%) at episode 48277, saving model...
11-29 05:46:46: New best reward 13.2 (+2.3%) at episode 50201, saving model...
11-29 05:48:07: New best reward 17.9 (+35.6%) at episode 53046, saving model...
11-29 05:48:24: New best reward 24.8 (+38.5%) at episode 53796, saving model...
11-29 05:50:38: New best reward 27.6 (+11.3%) at episode 59020, saving model...
11-29 05:53:58: New best reward 31.9 (+15.6%) at episode 67029, saving model...
11-29 05:57:28: New best reward 32.7 (+2.5%) at episode 74995, saving model...
11-29 05:57:43: New best reward 36.4 (+11.3%) at episode 75493, saving model...
11-29 05:58:48: New best reward 40.9 (+12.4%) at episode 77852, saving model...
11-29 05:59:32: New best reward 46.8 (+14.4%) at episode 79319, saving model...
11-29 06:12:28: New best reward 50.4 (+7.7%) at episode 105279, saving model...
11-29 06:17:34: New best reward 54.9 (+8.9%) at episode 114711, saving model...
11-29 06:31:19: New best reward 59.9 (+9.1%) at episode 138050, saving model...
```

```
11-29 06:31:19: New best reward 59.9 (+9.1%) at episode 138050, saving model...
```

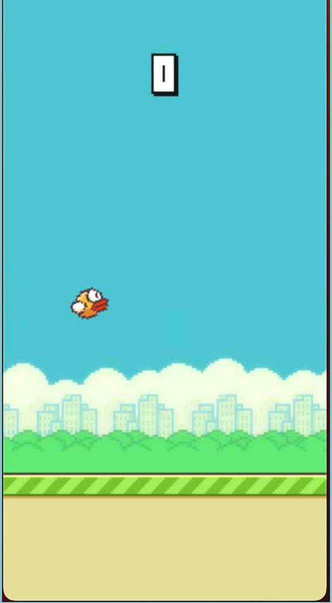
# Discount Factor : 0.90

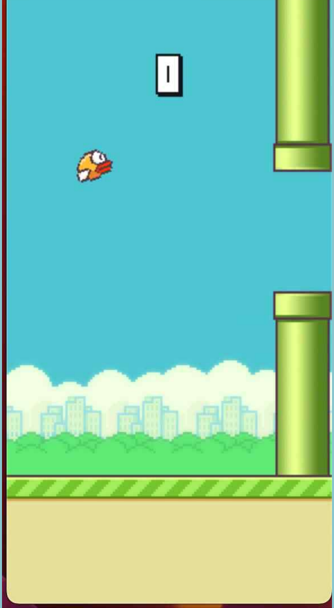


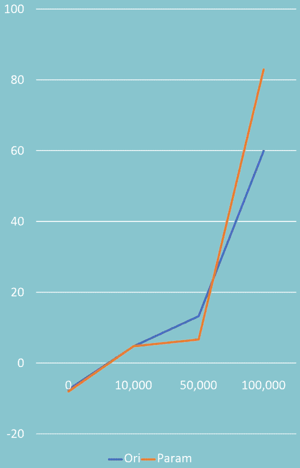
```
11-29 15:23:02: Training starting...
11-29 15:23:06: New best reward -8.1 (-100.0%) at episode 0, saving model...
11-29 15:23:06: New best reward -6.9 (-14.8%) at episode 1, saving model...
11-29 15:23:07: New best reward -5.1 (-26.1%) at episode 41, saving model...
11-29 15:23:08: New best reward -4.5 (-11.8%) at episode 91, saving model...
11-29 15:23:08: New best reward -2.7 (-40.0%) at episode 93, saving model...
11-29 15:23:11: New best reward -0.3 (-88.9%) at episode 342, saving model...
11-29 15:23:20: New best reward 1.5 (-600.0%) at episode 777, saving model...
11-29 15:23:28: New best reward 3.3 (+120.0%) at episode 1244, saving model...
11-29 15:23:29: New best reward 3.9 (+18.2%) at episode 1306, saving model...
11-29 15:25:49: New best reward 4.1 (+5.1%) at episode 8825, saving model...
11-29 15:26:40: New best reward 4.7 (+14.6%) at episode 11103, saving model...
11-29 15:28:08: New best reward 4.9 (+4.3%) at episode 14730, saving model...
11-29 15:28:40: New best reward 6.4 (+30.6%) at episode 15943, saving model...
11-29 15:47:39: New best reward 6.6 (+3.1%) at episode 50907, saving model...
11-29 15:48:59: New best reward 8.4 (+27.3%) at episode 52714, saving model...
11-29 15:49:27: New best reward 8.6 (+2.4%) at episode 53407, saving model...
11-29 15:49:57: New best reward 11.0 (+27.9%) at episode 54237, saving model...
11-29 15:50:13: New best reward 12.9 (+17.3%) at episode 54711, saving model...
11-29 15:50:52: New best reward 13.5 (+4.7%) at episode 55618, saving model...
11-29 15:51:02: New best reward 17.9 (+32.6%) at episode 55803, saving model...
11-29 15:51:39: New best reward 22.4 (+25.1%) at episode 56708, saving model...
11-29 15:52:44: New best reward 26.9 (+20.1%) at episode 58082, saving model...
11-29 15:56:32: New best reward 31.9 (+18.6%) at episode 62873, saving model...
11-29 15:57:24: New best reward 37.6 (+17.9%) at episode 63978, saving model...
11-29 16:02:33: New best reward 40.9 (+8.8%) at episode 70560, saving model...
11-29 16:17:01: New best reward 43.2 (+5.6%) at episode 83089, saving model...
11-29 16:22:24: New best reward 43.6 (+0.9%) at episode 87747, saving model...
11-29 16:25:46: New best reward 50.4 (+15.6%) at episode 90725, saving model...
11-29 16:33:53: New best reward 54.9 (+8.9%) at episode 97317, saving model...
11-29 16:38:18: New best reward 73.9 (+34.6%) at episode 100699, saving model...
11-29 17:05:16: New best reward 82.9 (+12.2%) at episode 119225, saving model...
```

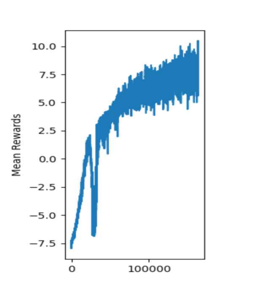


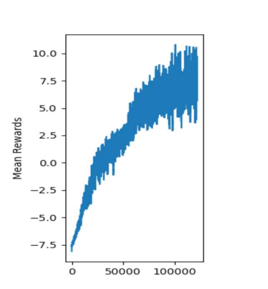
```
New best reward 82.9 (+12.2%) at episode 119225, saving model...
```


1


1





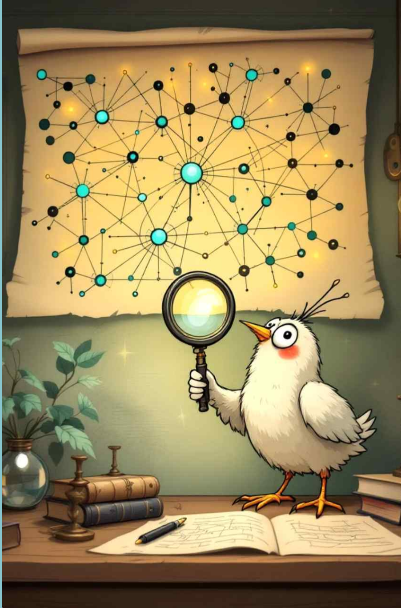


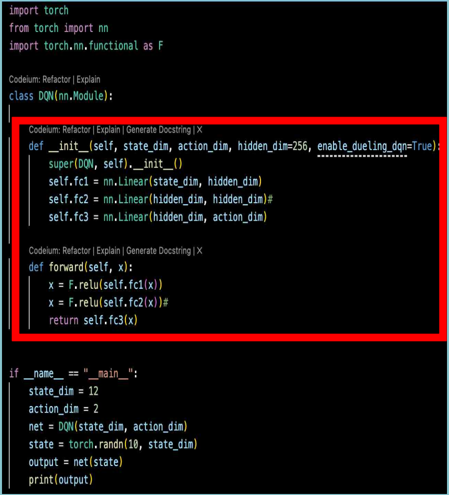
**Discount Factor : 0.99**
**Episode: 100,000**
**TIME taken:46m**
**Best Reward: 59.9**

**Discount Factor : 0.90**
**Episode: 100,000**
**TIME taken:1h 42m**
**Best Reward: 82.9**

# THIRD AIM

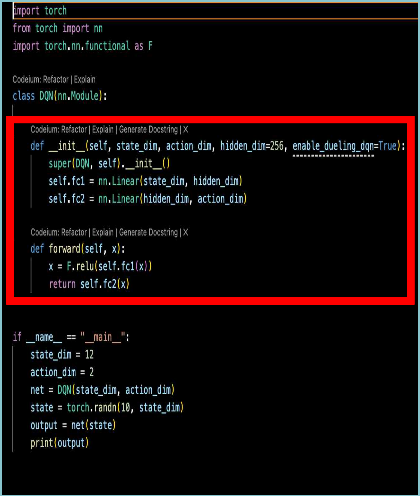
# Hidden Layer의 개수를 늘리면??





```
import torch
from torch import nn
import torch.nn.functional as F

class DQN(nn.Module):

    def __init__(self, state_dim, action_dim, hidden_dim=256, enable_dueling_dqn=True):
        super(DQN, self).__init__()
        self.fc1 = nn.Linear(state_dim, hidden_dim)
        self.fc2 = nn.Linear(hidden_dim, hidden_dim)#
        self.fc3 = nn.Linear(hidden_dim, action_dim)

    def forward(self, x):
        x = F.relu(self.fc1(x))
        x = F.relu(self.fc2(x))#
        return self.fc3(x)


if __name__ == "__main__":
    state_dim = 12
    action_dim = 2
    net = DQN(state_dim, action_dim)
    state = torch.randn(10, state_dim)
    output = net(state)
    print(output)
```

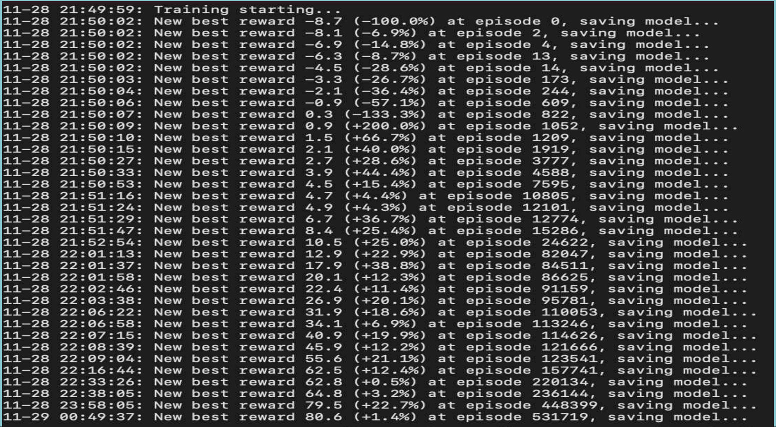
Codeium: Refactor | Explain
Codeium: Refactor | Explain | Generate Docstring | X
Codeium: Refactor | Explain | Generate Docstring | X



```
import torch
from torch import nn
import torch.nn.functional as F

class DQN(nn.Module):

    def __init__(self, state_dim, action_dim, hidden_dim=256, enable_dueling_dqn=True):
        super(DQN, self).__init__()
        self.fc1 = nn.Linear(state_dim, hidden_dim)
        self.fc2 = nn.Linear(hidden_dim, action_dim)

    def forward(self, x):
        x = F.relu(self.fc1(x))
        return self.fc2(x)


if __name__ == "__main__":
    state_dim = 12
    action_dim = 2
    net = DQN(state_dim, action_dim)
    state = torch.randn(10, state_dim)
    output = net(state)
    print(output)
```

Codeium: Refactor | Explain
Codeium: Refactor | Explain | Generate Docstring | X
Codeium: Refactor | Explain | Generate Docstring | X

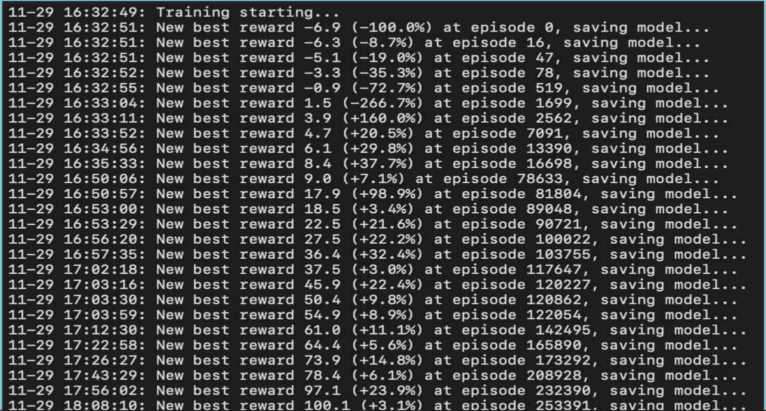
# Hidden Layer : 1



```
11-28 21:49:59: Training starting...
11-28 21:50:02: New best reward -8.7 (-100.0%) at episode 0, saving model...
11-28 21:50:02: New best reward -8.1 (-6.9%) at episode 2, saving model...
11-28 21:50:02: New best reward -6.9 (-14.8%) at episode 4, saving model...
11-28 21:50:02: New best reward -6.3 (-8.7%) at episode 13, saving model...
11-28 21:50:02: New best reward -4.5 (-28.6%) at episode 14, saving model...
11-28 21:50:03: New best reward -3.3 (-26.7%) at episode 173, saving model...
11-28 21:50:04: New best reward -2.1 (-36.4%) at episode 244, saving model...
11-28 21:50:06: New best reward -0.9 (-57.1%) at episode 609, saving model...
11-28 21:50:07: New best reward 0.3 (-133.3%) at episode 822, saving model...
11-28 21:50:09: New best reward 0.9 (+200.0%) at episode 1052, saving model...
11-28 21:50:10: New best reward 1.5 (+66.7%) at episode 1209, saving model...
11-28 21:50:15: New best reward 2.1 (+40.0%) at episode 1919, saving model...
11-28 21:50:27: New best reward 2.7 (+28.6%) at episode 3777, saving model...
11-28 21:50:33: New best reward 3.9 (+44.4%) at episode 4588, saving model...
11-28 21:50:53: New best reward 4.5 (+15.4%) at episode 7595, saving model...
11-28 21:51:16: New best reward 4.7 (+4.4%) at episode 10805, saving model...
11-28 21:51:24: New best reward 4.9 (+4.3%) at episode 12101, saving model...
11-28 21:51:29: New best reward 6.7 (+36.7%) at episode 12774, saving model...
11-28 21:51:47: New best reward 8.4 (+25.4%) at episode 15286, saving model...
11-28 21:52:54: New best reward 10.5 (+25.0%) at episode 24622, saving model...
11-28 22:01:13: New best reward 12.9 (+22.9%) at episode 82047, saving model...
11-28 22:01:37: New best reward 17.9 (+38.8%) at episode 84511, saving model...
11-28 22:01:58: New best reward 20.1 (+12.3%) at episode 86625, saving model...
11-28 22:02:46: New best reward 22.4 (+11.4%) at episode 91159, saving model...
11-28 22:03:38: New best reward 26.9 (+20.1%) at episode 95781, saving model...
11-28 22:06:22: New best reward 31.9 (+18.6%) at episode 110053, saving model...
11-28 22:06:58: New best reward 34.1 (+6.9%) at episode 113246, saving model...
11-28 22:07:15: New best reward 40.9 (+19.9%) at episode 114626, saving model...
11-28 22:08:39: New best reward 45.9 (+12.2%) at episode 121666, saving model...
11-28 22:09:04: New best reward 55.6 (+21.1%) at episode 123541, saving model...
11-28 22:16:44: New best reward 62.5 (+12.4%) at episode 157741, saving model...
11-28 22:33:26: New best reward 62.8 (+0.5%) at episode 220134, saving model...
11-28 22:38:05: New best reward 64.8 (+3.2%) at episode 236144, saving model...
11-28 23:58:05: New best reward 79.5 (+22.7%) at episode 448399, saving model...
11-29 00:49:37: New best reward 80.6 (+1.4%) at episode 531719, saving model...
```

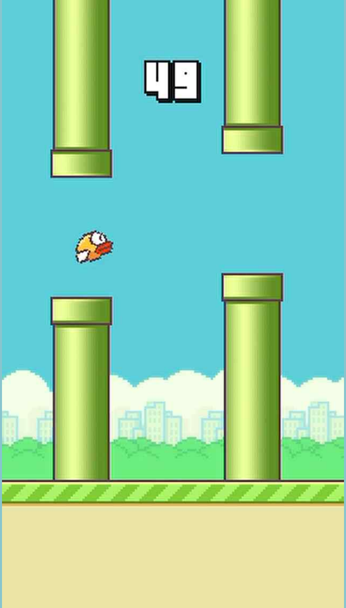
```
11-29 00:49:37: New best reward 80.6 (+1.4%) at episode 531719, saving model...
```

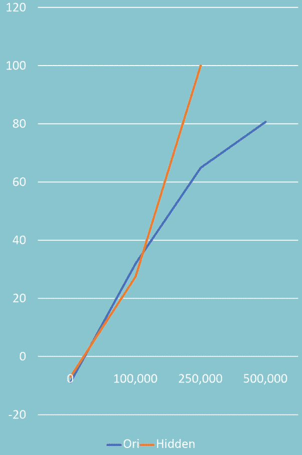
# Hidden Layer : 2

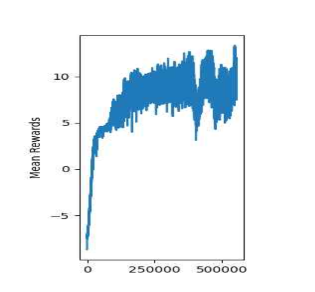


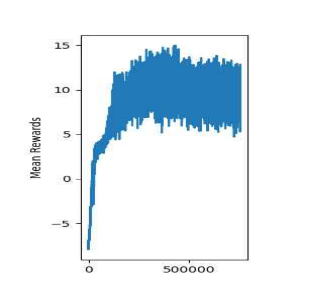
```
11-29 16:32:49: Training starting...
11-29 16:32:51: New best reward -6.9 (-100.0%) at episode 0, saving model...
11-29 16:32:51: New best reward -6.3 (-8.7%) at episode 16, saving model...
11-29 16:32:51: New best reward -5.1 (-19.0%) at episode 47, saving model...
11-29 16:32:52: New best reward -3.3 (-35.3%) at episode 78, saving model...
11-29 16:32:55: New best reward -0.9 (-72.7%) at episode 519, saving model...
11-29 16:33:04: New best reward 1.5 (-266.7%) at episode 1699, saving model...
11-29 16:33:11: New best reward 3.9 (+160.0%) at episode 2562, saving model...
11-29 16:33:52: New best reward 4.7 (+20.5%) at episode 7091, saving model...
11-29 16:34:56: New best reward 6.1 (+29.8%) at episode 13390, saving model...
11-29 16:35:33: New best reward 8.4 (+37.7%) at episode 16698, saving model...
11-29 16:50:06: New best reward 9.0 (+7.1%) at episode 78633, saving model...
11-29 16:50:57: New best reward 17.9 (+98.9%) at episode 81804, saving model...
11-29 16:53:00: New best reward 18.5 (+3.4%) at episode 89048, saving model...
11-29 16:53:29: New best reward 22.5 (+21.6%) at episode 90721, saving model...
11-29 16:56:20: New best reward 27.5 (+22.2%) at episode 100022, saving model...
11-29 16:57:35: New best reward 36.4 (+32.4%) at episode 103755, saving model...
11-29 17:02:18: New best reward 37.5 (+3.0%) at episode 117647, saving model...
11-29 17:03:16: New best reward 45.9 (+22.4%) at episode 120227, saving model...
11-29 17:03:30: New best reward 50.4 (+9.8%) at episode 120862, saving model...
11-29 17:03:59: New best reward 54.9 (+8.9%) at episode 122054, saving model...
11-29 17:12:30: New best reward 61.0 (+11.1%) at episode 142495, saving model...
11-29 17:22:58: New best reward 64.4 (+5.6%) at episode 165890, saving model...
11-29 17:26:27: New best reward 73.9 (+14.8%) at episode 173292, saving model...
11-29 17:43:29: New best reward 78.4 (+6.1%) at episode 208928, saving model...
11-29 17:56:02: New best reward 97.1 (+23.9%) at episode 232390, saving model...
11-29 18:08:10: New best reward 100.1 (+3.1%) at episode 253391, saving model...
```

```
11-29 18:08:10: New best reward 100.1 (+3.1%) at episode 253391, saving model...
```


49





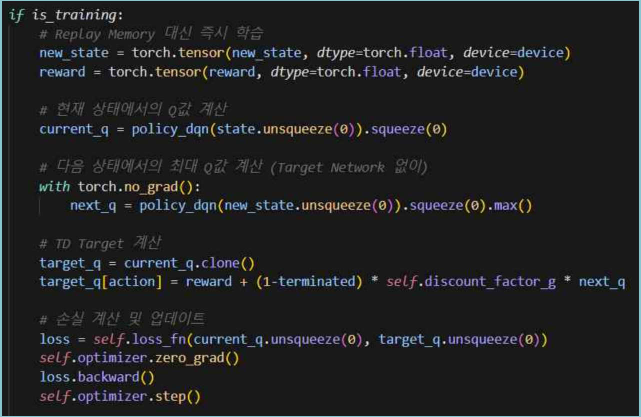


**Hidden Layer : 1**
**Episode : 500,000**
**Best Reward : 80.6**
**Time taken : 3h**

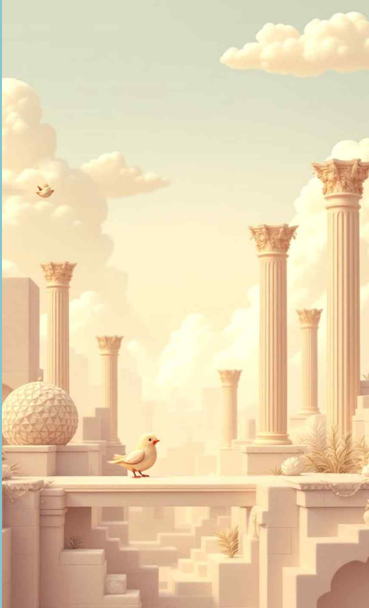
**Hidden Layer : 2**
**Episode : 250,000**
**Best Reward : 100.1**
**Time taken : 1h 36m**

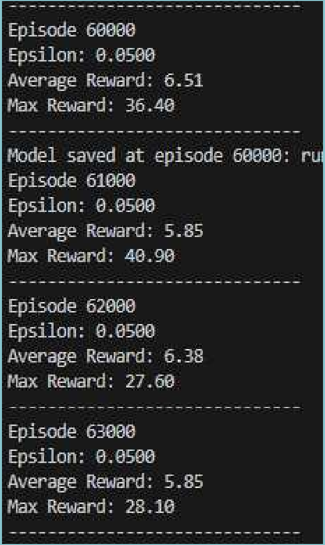
# FORTH AIM

# Naive DQN

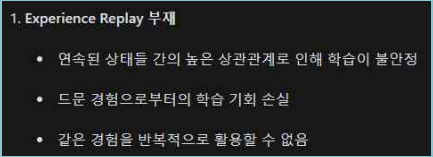


```python
if is_training:
    # Replay Memory 대신 즉시 학습
    new_state = torch.tensor(new_state, dtype=torch.float, device=device)
    reward = torch.tensor(reward, dtype=torch.float, device=device)

    # 현재 상태에서의 Q값 계산
    current_q = policy_dqn(state.unsqueeze(0)).squeeze(0)

    # 다음 상태에서의 최대 Q값 계산 (Target Network 없이)
    with torch.no_grad():
        next_q = policy_dqn(new_state.unsqueeze(0)).squeeze(0).max()

    # TD Target 계산
    target_q = current_q.clone()
    target_q[action] = reward + (1-terminated) * self.discount_factor_g * next_q

    # 손실 계산 및 업데이트
    loss = self.loss_fn(current_q.unsqueeze(0), target_q.unsqueeze(0))
    self.optimizer.zero_grad()
    loss.backward()
    self.optimizer.step()
```



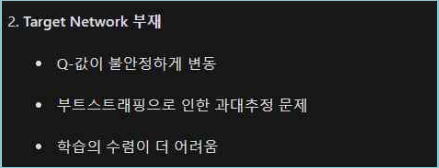


```
--------------------------------
Episode 60000
Epsilon: 0.0500
Average Reward: 6.51
Max Reward: 36.40
--------------------------------
Model saved at episode 60000: ru
Episode 61000
Epsilon: 0.0500
Average Reward: 5.85
Max Reward: 40.90
--------------------------------
Episode 62000
Epsilon: 0.0500
Average Reward: 6.38
Max Reward: 27.60
--------------------------------
Episode 63000
Epsilon: 0.0500
Average Reward: 5.85
Max Reward: 28.10
--------------------------------
```



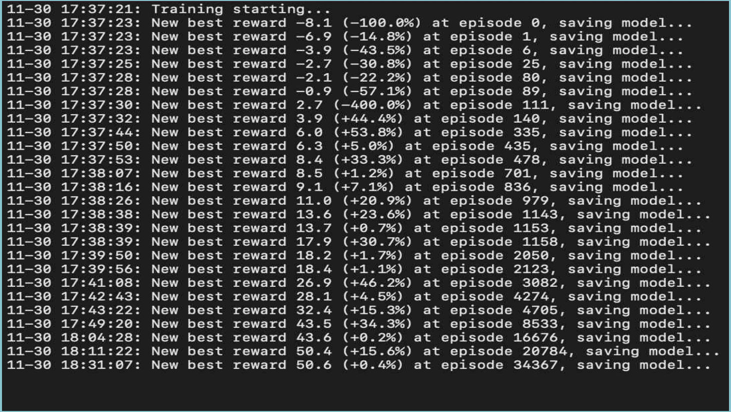
1. **Experience Replay 부재**

- 연속된 상태들 간의 높은 상관관계로 인해 학습이 불안정
- 드문 경험으로부터의 학습 기회 손실
- 같은 경험을 반복적으로 활용할 수 없음

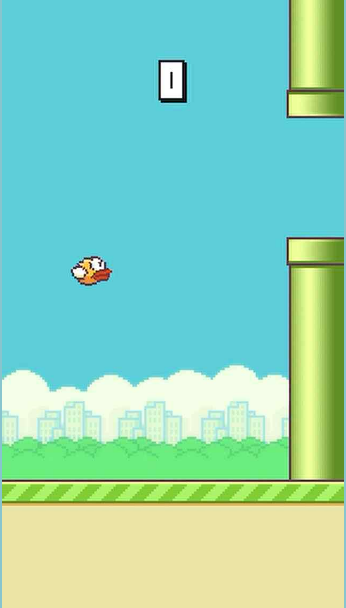


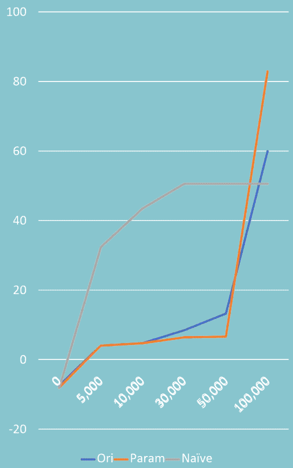
2. **Target Network 부재**

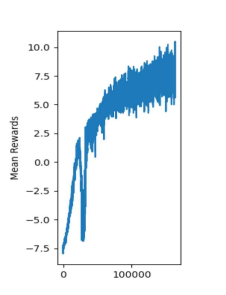
- Q-값이 불안정하게 변동
- 부트스트래핑으로 인한 과대추정 문제
- 학습의 수렴이 더 어려움



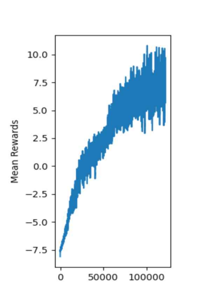
```
11-30 17:37:21: Training starting...
11-30 17:37:23: New best reward -8.1 (-100.0%) at episode 0, saving model...
11-30 17:37:23: New best reward -6.9 (-14.8%) at episode 1, saving model...
11-30 17:37:23: New best reward -3.9 (-43.5%) at episode 6, saving model...
11-30 17:37:25: New best reward -2.7 (-30.8%) at episode 25, saving model...
11-30 17:37:28: New best reward -2.1 (-22.2%) at episode 80, saving model...
11-30 17:37:28: New best reward -0.9 (-57.1%) at episode 89, saving model...
11-30 17:37:30: New best reward 2.7 (-400.0%) at episode 111, saving model...
11-30 17:37:32: New best reward 3.9 (+44.4%) at episode 140, saving model...
11-30 17:37:44: New best reward 6.0 (+53.8%) at episode 335, saving model...
11-30 17:37:50: New best reward 6.3 (+5.0%) at episode 435, saving model...
11-30 17:37:53: New best reward 8.4 (+33.3%) at episode 478, saving model...
11-30 17:38:07: New best reward 8.5 (+1.2%) at episode 701, saving model...
11-30 17:38:16: New best reward 9.1 (+7.1%) at episode 836, saving model...
11-30 17:38:26: New best reward 11.0 (+20.9%) at episode 979, saving model...
11-30 17:38:38: New best reward 13.6 (+23.6%) at episode 1143, saving model...
11-30 17:38:39: New best reward 13.7 (+0.7%) at episode 1153, saving model...
11-30 17:38:39: New best reward 17.9 (+30.7%) at episode 1158, saving model...
11-30 17:39:50: New best reward 18.2 (+1.7%) at episode 2050, saving model...
11-30 17:39:56: New best reward 18.4 (+1.1%) at episode 2123, saving model...
11-30 17:41:08: New best reward 26.9 (+46.2%) at episode 3082, saving model...
11-30 17:42:43: New best reward 28.1 (+4.5%) at episode 4274, saving model...
11-30 17:43:22: New best reward 32.4 (+15.3%) at episode 4705, saving model...
11-30 17:49:20: New best reward 43.5 (+34.3%) at episode 8533, saving model...
11-30 18:04:28: New best reward 43.6 (+0.2%) at episode 16676, saving model...
11-30 18:11:22: New best reward 50.4 (+15.6%) at episode 20784, saving model...
11-30 18:31:07: New best reward 50.6 (+0.4%) at episode 34367, saving model...
```


1

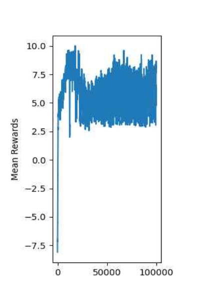




**1. ORIGINAL**
**Episode: 100,000**
**TIME taken : 46m**
**Best Reward : 59.9**



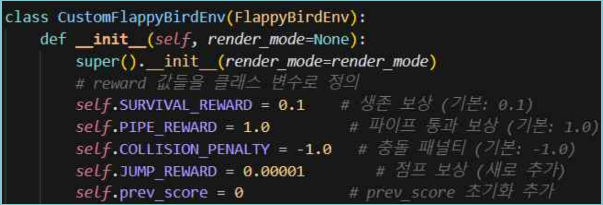
**discount factor : 0.90**
**Episode: 100,000**
**TIME taken :1h 42m**
**Best Reward : 82.9**



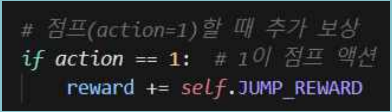
**3. NAÏVE**
**Episode: 100,000**
**TIME taken : 54m**
**Best Reward :  50.6**

# FIFTH AIM

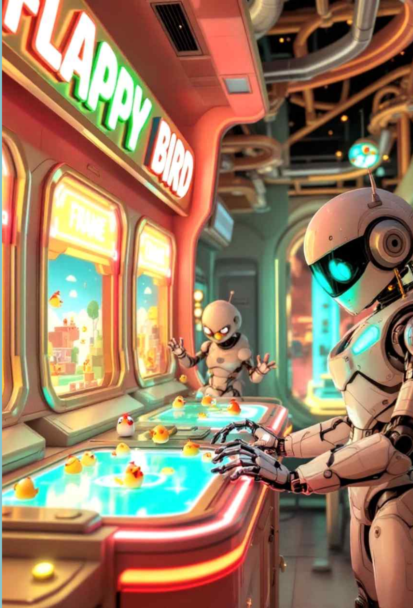
# Custom Env

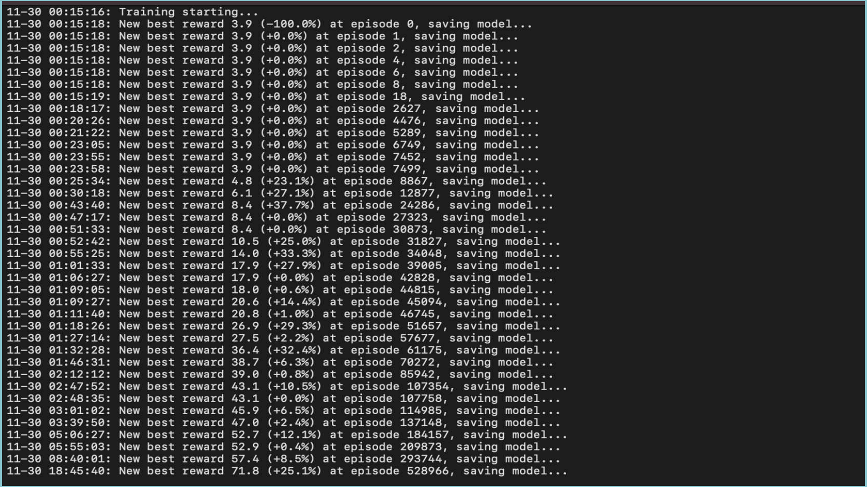


```python
class CustomFlappyBirdEnv(FlappyBirdEnv):
    def __init__(self, render_mode=None):
        super().__init__(render_mode=render_mode)
        # reward 값들을 클래스 변수로 정의
        self.SURVIVAL_REWARD = 0.1    # 생존 보상 (기본: 0.1)
        self.PIPE_REWARD = 1.0          # 파이프 통과 보상 (기본: 1.0)
        self.COLLISION_PENALTY = -1.0  # 충돌 패널티 (기본: -1.0)
        self.JUMP_REWARD = 0.00001       # 점프 보상 (새로 추가)
        self.prev_score = 0             # prev_score 초기화 추가
```

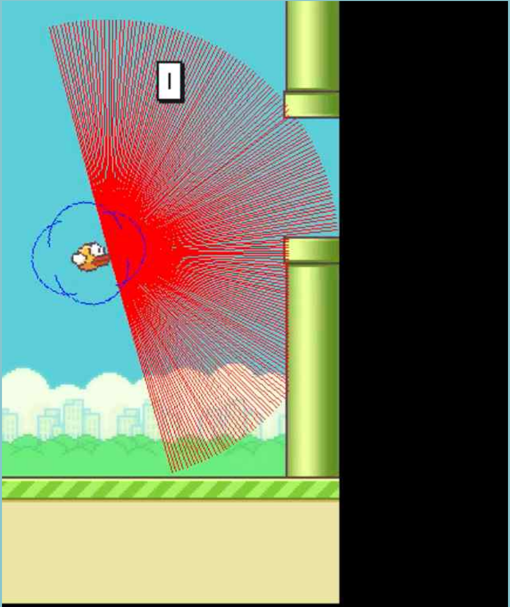


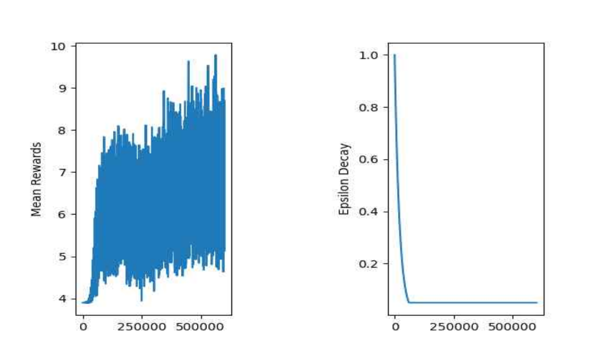
```python
# 점프(action=1)할 때 추가 보상
if action == 1:  # 1이 점프 액션
    reward += self.JUMP_REWARD
```





```
11-30 00:15:16: Training starting...
11-30 00:15:18: New best reward 3.9 (-100.0%) at episode 0, saving model...
11-30 00:15:18: New best reward 3.9 (+0.0%) at episode 1, saving model...
11-30 00:15:18: New best reward 3.9 (+0.0%) at episode 2, saving model...
11-30 00:15:18: New best reward 3.9 (+0.0%) at episode 4, saving model...
11-30 00:15:18: New best reward 3.9 (+0.0%) at episode 6, saving model...
11-30 00:15:18: New best reward 3.9 (+0.0%) at episode 8, saving model...
11-30 00:15:19: New best reward 3.9 (+0.0%) at episode 18, saving model...
11-30 00:18:17: New best reward 3.9 (+0.0%) at episode 2627, saving model...
11-30 00:20:26: New best reward 3.9 (+0.0%) at episode 4476, saving model...
11-30 00:21:22: New best reward 3.9 (+0.0%) at episode 5289, saving model...
11-30 00:23:05: New best reward 3.9 (+0.0%) at episode 6749, saving model...
11-30 00:23:55: New best reward 3.9 (+0.0%) at episode 7452, saving model...
11-30 00:23:58: New best reward 3.9 (+0.0%) at episode 7499, saving model...
11-30 00:25:34: New best reward 4.8 (+23.1%) at episode 8867, saving model...
11-30 00:30:18: New best reward 6.1 (+27.1%) at episode 12877, saving model...
11-30 00:43:40: New best reward 8.4 (+37.7%) at episode 24286, saving model...
11-30 00:47:17: New best reward 8.4 (+0.0%) at episode 27323, saving model...
11-30 00:51:33: New best reward 8.4 (+0.0%) at episode 30873, saving model...
11-30 00:52:42: New best reward 10.5 (+25.0%) at episode 31827, saving model...
11-30 00:55:25: New best reward 14.0 (+33.3%) at episode 34048, saving model...
11-30 01:01:33: New best reward 17.9 (+27.9%) at episode 39005, saving model...
11-30 01:06:27: New best reward 17.9 (+0.0%) at episode 42828, saving model...
11-30 01:09:05: New best reward 18.0 (+0.6%) at episode 44815, saving model...
11-30 01:09:27: New best reward 20.6 (+14.4%) at episode 45094, saving model...
11-30 01:11:40: New best reward 20.8 (+1.0%) at episode 46745, saving model...
11-30 01:18:26: New best reward 26.9 (+29.3%) at episode 51657, saving model...
11-30 01:27:14: New best reward 27.5 (+2.2%) at episode 57677, saving model...
11-30 01:32:28: New best reward 36.4 (+32.4%) at episode 61175, saving model...
11-30 01:46:31: New best reward 38.7 (+6.3%) at episode 70272, saving model...
11-30 02:12:12: New best reward 39.0 (+0.8%) at episode 85942, saving model...
11-30 02:47:52: New best reward 43.1 (+10.5%) at episode 107354, saving model...
11-30 02:48:35: New best reward 43.1 (+0.0%) at episode 107758, saving model...
11-30 03:01:02: New best reward 45.9 (+6.5%) at episode 114985, saving model...
11-30 03:39:50: New best reward 47.0 (+2.4%) at episode 137148, saving model...
11-30 05:06:27: New best reward 52.7 (+12.1%) at episode 184157, saving model...
11-30 05:55:03: New best reward 52.9 (+0.4%) at episode 209873, saving model...
11-30 08:40:01: New best reward 57.4 (+8.5%) at episode 293744, saving model...
11-30 18:45:40: New best reward 71.8 (+25.1%) at episode 528966, saving model...
```


1



**Custom Flappybird**

**Best Reward : 71.8**

**Episode : 500,000**

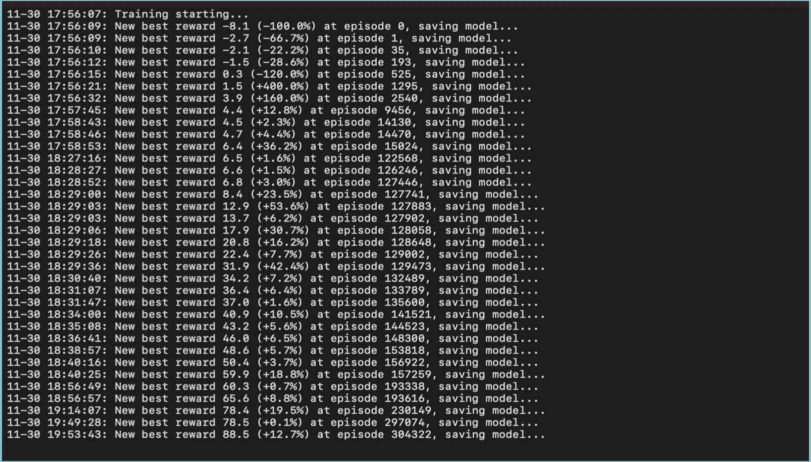
**Time taken : 18h 30m**

How about ?

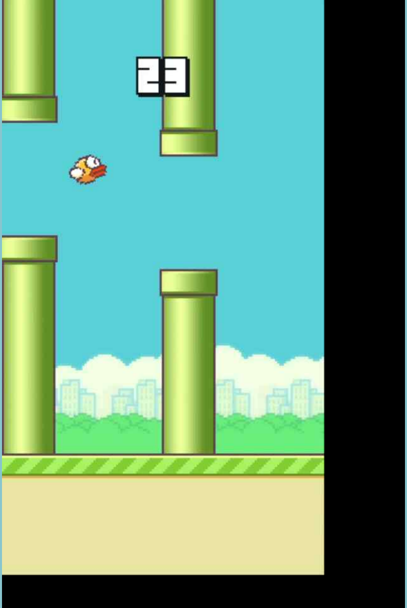
Hidden Layer 2개

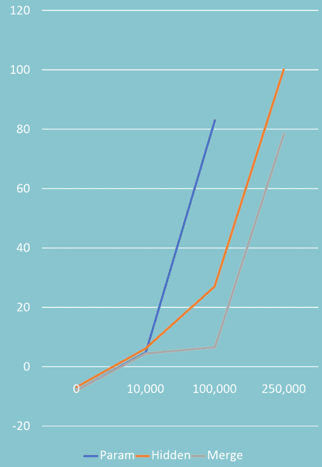
+

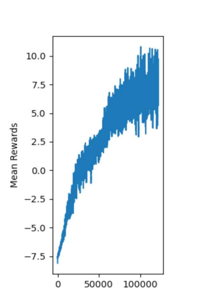
Discount Factor 0.90

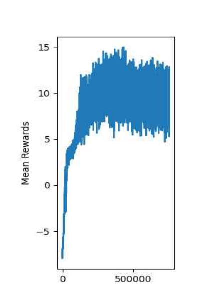


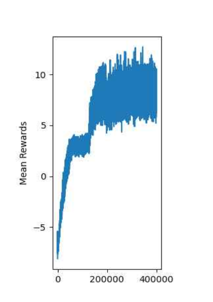
```
11-30 17:56:07: Training starting...
11-30 17:56:09: New best reward -8.1 (-100.0%) at episode 0, saving model...
11-30 17:56:09: New best reward -2.7 (-66.7%) at episode 1, saving model...
11-30 17:56:10: New best reward -2.1 (-22.2%) at episode 35, saving model...
11-30 17:56:12: New best reward -1.5 (-28.6%) at episode 193, saving model...
11-30 17:56:15: New best reward 0.3 (-120.0%) at episode 525, saving model...
11-30 17:56:21: New best reward 1.5 (+400.0%) at episode 1295, saving model...
11-30 17:56:32: New best reward 3.9 (+160.0%) at episode 2540, saving model...
11-30 17:57:45: New best reward 4.4 (+12.8%) at episode 9456, saving model...
11-30 17:58:43: New best reward 4.5 (+2.3%) at episode 14130, saving model...
11-30 17:58:46: New best reward 4.7 (+4.4%) at episode 14470, saving model...
11-30 17:58:53: New best reward 6.4 (+36.2%) at episode 15024, saving model...
11-30 18:27:16: New best reward 6.5 (+1.6%) at episode 122568, saving model...
11-30 18:28:27: New best reward 6.6 (+1.5%) at episode 126246, saving model...
11-30 18:28:52: New best reward 6.8 (+3.0%) at episode 127446, saving model...
11-30 18:29:00: New best reward 8.4 (+23.5%) at episode 127741, saving model...
11-30 18:29:03: New best reward 12.9 (+53.6%) at episode 127883, saving model...
11-30 18:29:03: New best reward 13.7 (+6.2%) at episode 127902, saving model...
11-30 18:29:06: New best reward 17.9 (+30.7%) at episode 128058, saving model...
11-30 18:29:18: New best reward 20.8 (+16.2%) at episode 128648, saving model...
11-30 18:29:26: New best reward 22.4 (+7.7%) at episode 129002, saving model...
11-30 18:29:36: New best reward 31.9 (+42.4%) at episode 129473, saving model...
11-30 18:30:40: New best reward 34.2 (+7.2%) at episode 132489, saving model...
11-30 18:31:07: New best reward 36.4 (+6.4%) at episode 133789, saving model...
11-30 18:31:47: New best reward 37.0 (+1.6%) at episode 135600, saving model...
11-30 18:34:00: New best reward 40.9 (+10.5%) at episode 141521, saving model...
11-30 18:35:08: New best reward 43.2 (+5.6%) at episode 144523, saving model...
11-30 18:36:41: New best reward 46.0 (+6.5%) at episode 148300, saving model...
11-30 18:38:57: New best reward 48.6 (+5.7%) at episode 153818, saving model...
11-30 18:40:16: New best reward 50.4 (+3.7%) at episode 156922, saving model...
11-30 18:40:25: New best reward 59.9 (+18.8%) at episode 157259, saving model...
11-30 18:56:49: New best reward 60.3 (+0.7%) at episode 193338, saving model...
11-30 18:56:57: New best reward 65.6 (+8.8%) at episode 193616, saving model...
11-30 19:14:07: New best reward 78.4 (+19.5%) at episode 230149, saving model...
11-30 19:49:28: New best reward 78.5 (+0.1%) at episode 297074, saving model...
11-30 19:53:43: New best reward 88.5 (+12.7%) at episode 304322, saving model...
```


23









**discount factor : 0.90**
Episode: 100,000
Best Reward : 82.9
TIME taken :1h 42m

**Hidden layer : 2**
Episode : 250,000
Best Reward : 100.1
Time taken : 1h 36m

**Merging**
**Episode : 300,000**
**Best Reward : 88.5**
**Time taken :1h57m**

**Hidden layer 2개**

네트워크의 표현력이 증가

더 복잡한 패턴을 학습

**Discount factor 0.90**

더 즉각적인 보상에 집중

**이 두 가지가 결합되면, 복잡한 네트워크가**
**단기적 보상에 과적합(overfitting) 될 수 있다**

이런 현상은 강화학습에서 흔히 발생하는 "각각은 좋았지만 함께하면 안 좋은" 경우의 예시입니다.

# Discussion

Python

Mean 값 추적

Custom Env 이해

Grid Search