PREGUNTAS TIPO TEST

1. En el algoritmo Actor-Crític, la acción seleccionada en el entrenamiento viene dada por:

- **El Actor**
- El Crític
- El Actor y el crític
- Ninguno de los otros

2. Respecto al factor gamma visto en la asignatura, marque las afirmaciones correctas:

- **Un valor bajo enfatiza recompensas tempranas.**
- Un valor alto favorece la exploración.
- Un valor alto penaliza recompensas tardías.
- Se utiliza para ponderar la exploración en acciones futuras.

3. Cuál de los siguientes elementos se puede definir como "la estimación de la recompensa esperada a futuro dado un estado, y siguiendo una policy dada".

- Discounted Rewards
- **Función Value**
- Función Q
- Función Advantadge

4. En los métodos de policy gradients, la exploración viene dada por:

- El uso de una variable épsilon.
- **Un muestreo de la distribución de probabilidad de las acciones.**
- Incorporando ruido en la distribución de acciones con correlación temporal.
- Un muestreo de la salida lineal del modelo de la policy.

5. Indique cuáles de los siguientes métodos vistos en la asignatura permiten modelar espacios de acciones continuas:

- **DDPG**
- **PPO**
- AlphaGo
- DQN

6. En el contexto de soluciones de aprendizaje por refuerzo basadas en modelo, y en particular AlphaGo, las etapas secuenciales del algoritmo Montecarlo Tree Search (MCTS) son:

- **Selección - Expansión - Evaluación – Backup**
- Evaluación – Expansión – Backup – Selección
- Selección - Evaluación - Expansión – Feedforward
- Evaluación – Backup – Expansión – Feedforward

7. En su definición base, una Transición está compuesta de:

- Estado, Acción
- Estado, Acción, Recompensa.
- Acción, Recompensa.
- **Estado, Acción, Recompensa, Siguientes Estado.**

8. La arquitectura de Deep Learning más utilizada hoy en día en algoritmos de aprendizaje por refuerzo es:

- **Arquitectura Convolucional.**
- Transformers.
- Arquitectura Feed-Forward.
- Arquitectura Recurrente.

9. Cuando trabajamos en un enfoque model-based, el modelo se refiere a:

- Una arquitectura CNN que el agente usa durante las iteraciones.
- El modelo que utiliza el agente para preprocesar los datos de entrada.
- **El modelo que define las dinámicas del entorno y que el agente utiliza durante el proceso de aprendizaje.**
- El modelo que utiliza el agente para estimar qué acción seleccionar.

PREGUNTAS TEST AVANZADAS

1. En el algoritmo de Actor-Critic (puede haber varias respuestas correctas):

- El Critic se encarga de estimar la probabilidad de la mejor acción del estado actual.
- **El Critic estima el vale del estado actual.**
- **El Actor se encarga de estimar la distribución de probabilidades de las acciones.**
- El Actor se encarga de estimar la recompensa esperada para cada acción.

2. En el algoritmo A2C (puede haber varias respuestas correctas):

- **Es un algoritmo multi-proceso.**
- **En la función de coste, usa como factor de relevancia la función de ventaja.**
- Ejecuta una actualización del modelo de manera asíncrona.
- **Es un algoritmo on-policy.**

3. Respecto a la policy seguida en DQN (puede haber varias respuestas correctas):

- **Sigue una política voraz (o e-greedy) respecto a la función Q estimada.**
- La acción en cada estado se selecciona a partir del value estimado.
- Se favorece la exploración en entrenamiento mediante una épsilon creciente.
- **La calidad de la policy depende de la precisión en la estimación de la función Q.**

4. Respecto al algoritmo Deep Q-Networks, indique las afirmaciones correctas:

- **Se trata de un algoritmo off-policy.**
- La policy aprendida cumple la propiedad de recursividad.
- **La exploración se lleva a cabo mediante una política e-greedy.**
- **La función Q se optimiza a partir de la predicción de discounted rewards.**

5. En el algoritmo A3C (puede haber varias respuestas correctas):

- **Es un algoritmo multi-proceso.**
- **En la función de coste, usa como factor de relevancia la función de ventaja.**
- **Ejecuta una actualización del modelo de manera asíncrona.**
- **Es un algoritmo on-policy.**

6. En los algoritmos A2C y A3C (puede haber varias respuestas correctas):

- Se utiliza una target network para estimar el value.
- Son algoritmos multi-agente y multi-proceso.
- Resuelven la ineficacia de las muestras recolectadas.
- **Favorecen la exploración de trayectorias heterogéneas.**

PREGUNTAS TIPO DESARROLLO

1. Dado el algoritmo que se presenta en la imagen: (i) indique a qué familia de métodos pertenece, (ii) describa brevemente cómo modificaría el mismo en el caso de tener una interacción agente-entorno con trayectorias infinitas, y (iii) indique cuál es el papel del parámetro b. (1,5 punto)

Initialize policy parameter θ, baseline b
for iteration$=1, 2, \ldots$ **do**
  Collect a set of trajectories by executing the current policy
  At each timestep in each trajectory, compute
   the *return* $R_t = \sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'}$, and
   the *advantage estimate* $\hat{A}_t = R_t - b(s_t)$.
  Re-fit the baseline, by minimizing $\|b(s_t) - R_t\|^2$,
   summed over all trajectories and timesteps.
  Update the policy, using a policy gradient estimate $\hat{g}$,
   which is a sum of terms $\nabla_\theta \log \pi(a_t \mid s_t, \theta)\hat{A}_t$
end for

2. Dado el código parcial de un agente-entorno entrenado mediante deep reinforcement learning que se muestra a continuación, indique brevemente: (i) Qué tipo de algoritmo se ha empleado para entrenar, (ii) Qué tipo de estrategia sigue el agente durante la inferencia, (iii) Qué es la función phi y qué datos toma como entrada la DNN del agente para hacer el forward. (1,5 punto)

```
import torch
import gym

# DNN from agent
class DNN(torch.nn.Module):
    def __init__(self):
        super(DNN, self).__init__()
        self.conv1 = torch.nn.Conv2d(6, 32, 8, stride=4)
        self.conv2 = torch.nn.Conv2d(32, 64, 4, stride=2)
        self.conv3 = torch.nn.Conv2d(64, 64, 3, stride=1)
        self.fc1 = torch.nn.Linear(7 * 7 * 64, 512)
        self.out = torch.nn.Linear(512, 4)

    def forward(self, x):
        x = torch.nn.functional.relu(self.conv1(x))
        x = torch.nn.functional.relu(self.conv2(x))
        x = torch.nn.functional.relu(self.conv3(x))
        x = x.view(-1, 7 * 7 * 64)
        x = torch.nn.functional.relu(self.fc1(x))
        out = self.out(x)
        return out

# Cargar el agente
model = DNN()
model.load_state_dict(torch.load("path_to_best_model.pth"))
```

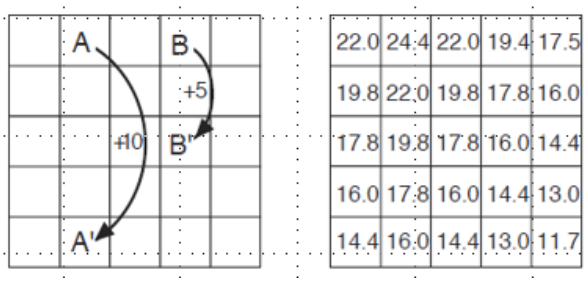
```
# Definición del entorno
env = gym.make(ENV_NAME)
o = env.reset()
s = None

# Interacción entorno-agente
play = True
while play:
  s = phi(s, o)
  out = model(s)
  action = torch.argmax(out)
  o, r, done, _ = env.step(action.item())
  if done:
    play=False
```

OTRAS PREGUNTAS PARA REFORZAR CONCEPTOS

1. Se quiere optimizar un agente utilizando DQN. A partir del entorno indicado, y un agente a mitad de entrenamiento. Dado que el espacio de estados es finito, se ha realizado una evaluación de nuestra función. Asimismo, se fija un discount factor, $\gamma = 0'9$.

1. **Comenzando en $s = [2, 2]$, y siguiendo una política voraz, calcular las pérdidas en el siguiente step.**
2. **Comenzando en $s = [2, 2]$, y siguiendo una e-greedy, que nos lleva a la derecha, calcular las pérdidas en el siguiente step.**



- Las **celdas** corresponden con los **estados** del entorno. $S = [Fila, Columna]$.
- El espacio de acciones es: $A = [izq, der, arriba, abajo]$.
- La recompensa para cada acción-estado es:
  - **+10** para la transición de **A a A'**
  - **+5** para la transición de **B a B'**
  - **-1** si el movimiento se sale del *grid.*
  - **0** en cualquier otro caso.
- Una vez en A o B, la unica acción possible es moverse a A' o B', respectivamente

$$q(s, a) \rightarrow r + \gamma max_a q(s', a)$$

$$\boldsymbol{L = (r + \gamma max_a q(s', a)) - q(s, a))^2}$$

2. Identifique cuáles son el entorno, el agente, y el algoritmo de aprendizaje por refuerzo en el siguiente algoritmo de aprendizaje por refuerzo a partir de feedback humano. (https://huggingface.co/blog/rlhf)