Assignments:

1. **Optional** Implement the Monte-Carlo Policy Gradient (REINFORCE) algorithm in Python and test it by checking that you recover the closed-form solution of the Discrete-Time Asset-Allocation example (single risky asset with no consumption before terminal date). The lecture slides have the pseudo-code for this algorithm.

2. **Optional** Implement the ACTOR-CRITIC-ELIGIBILITY-TRACES Policy Gradient algorithm in Python and test it by checking that you recover the closed-form solution of the Discrete-Time Asset-Allocation example (single risky asset with no consumption before terminal date). The lecture slides have the pseudo-code for this algorithm.

3. Assume we have a finite action space $\mathcal{A}$. Let $\phi(s,a) = (\phi_1(s,a), \phi_2(s,a), \ldots, \phi_m(s,a))$ be the features vector for any $s \in \mathcal{N}, a \in \mathcal{A}$. Let $\theta = (\theta_1, \theta_2, \ldots, \theta_m)$ be an $m$-vector of parameters. Let the action probabilities conditional on a given state $s$ and given parameter vector $\theta$ be defined by the softmax function on the linear combination of features: $\pi(s,a;\theta) = e^{\phi(s,a)^T \cdot \theta} / \sum_{b \in \mathcal{A}} e^{\phi(s,b)^T \cdot \theta}$, i.e.,

   $$\pi(s,a;\theta) = \frac{e^{\phi(s,a)^T \cdot \theta}}{\sum_{b \in \mathcal{A}} e^{\phi(s,b)^T \cdot \theta}}$$

   - Evaluate the score function $\nabla_{\theta} \log \pi(s,a;\theta)$
   - Construct the Action-Value function approximation $Q(s,a;w)$ so that the following key constraint of the Compatible Function Approximation Theorem (for Policy Gradient) is satisfied:
     $$\nabla_w Q(s,a;w) = \nabla_{\theta} \log \pi(s,a;\theta)$$

     where $w$ defines the parameters of the function approximation of the Action-Value function.

   - Show that $Q(s,a;w)$ has zero mean for any state $s$, i.e. show that
     $$\mathbb{E}_{\pi}[Q(s,a;w)] \text{ defined as } \sum_{a \in \mathcal{A}} \pi(s,a;\theta) \cdot Q(s,a;w) = 0 \text{ for all } s \in \mathcal{N}$$