SAC Algorithm

Input: θ_1, θ_2, ϕ ▷ Initial critic and actor networks parameters
$\bar{\theta}_1 \leftarrow \theta_1$, $\bar{\theta}_2 \leftarrow \theta_2$ ▷ Initialize target network weights
$D \leftarrow \emptyset$ ▷ Initialize an empty replay buffer
for each iteration **do**
    for each environment step **do**
        $a_t \sim \pi_\phi(a_t|s_t)$ ▷ Sample action from the policy
        $s_{t+1} \sim p(s_{t+1}|s_t, a_t)$ ▷ Receive transition from the environment
        $D \leftarrow D \cup \{(s_t, a_t, R(s_t, a_t), s_{t+1})\}$ ▷ Store the transition in the replay buffer
    end for
    for each gradient step **do**
        $\theta_i \leftarrow \theta_i - \lambda_Q \hat{\nabla}_{\theta_i} J_Q(\theta_i)$ for $i \in \{1, 2\}$ ▷ Update critic networks Q-functions
        $\phi \leftarrow \phi - \lambda_\pi \hat{\nabla}_\phi J_\pi(\phi)$ ▷ Update actor network policy
        $\alpha \leftarrow \alpha - \lambda_\alpha \hat{\nabla}_\alpha J(\alpha)$ ▷ Adjust temperature
        $\bar{\theta}_i \leftarrow \tau\theta_i + (1-\tau)\bar{\theta}_i$ for $i \in \{1, 2\}$ ▷ Update target network weights
    end for
end for
Output: θ_1, θ_2, ϕ ▷ Optimized critic and actor networks parameters