Algorithm: QMIX Training

Initialize: θ and θ^- the parameters of individual and target Q-networks Q_i, ϕ and ϕ^- the parameters of the mixing and target-mixing network m.
Initialize replay buffer $\mathcal{D}$ // $(episode_1, episode_2, \dots)$
while $t < T$ **do**
    $current\_episode = \{\}$
    while $\mathbf{o}_t$ is not *done* **do**
        Collect observations $\{o_1^t, \dots, o_n^t\}$ and state $\mathbf{s}_t$
        for each agent i **do**
            With probability ϵ, select random action a_i^t
            otherwise select $a_i^t = \arg\max_{a_i} Q_i(o_i^t, a_i)$
        end for
        Execute joint action $\mathbf{a}^t = (a_1^t, \dots, a_n^t)$
        Collect r^t and $done^t$
        Store $(\mathbf{s}_t, \mathbf{o}^t, \mathbf{a}^t, r^t, done^t,)$ in $current\_episode$
    end while
    Store $current\_episode$ in the replay buffer $\mathcal{D}$
    if t is a training step **then**
        Sample batch of episodes

$$\mathcal{B} = \{\{\mathbf{s}^{t,b}, \mathbf{o}^{t,b}, \mathbf{a}^{t,b}, r^{t,b}, done^{t,b}, \mathbf{s'}^{t,b}, \mathbf{o'}^{t,b}\}_{t=1\dots L^b}\}_{b=1,\dots,|\mathcal{B}|}$$

        Set the targets

$$y^{t,b} = r^{t,b} + \gamma(1 - done^{t,b}) \times \max_{(a_1,\dots,a_n)} m(\mathbf{s'}^{t,b}, Q_1(\mathbf{o}_1^{t,b}, a_1; \theta^-), \dots, Q_n(\mathbf{o}_n^{t,b}, a_n; \theta^-); \phi^-)$$

        Perform a gradient descent using:

$$\mathcal{L}(\theta, \phi) = \frac{1}{|\mathcal{B}|} \sum_b \frac{1}{L^b} \sum_t \left(y^{t,b} - Q^{tot}(\mathbf{s}^{t,b}, \mathbf{o}^{t,b}, \mathbf{a}^{t,b}; \theta, \phi)\right)^2$$

        Every C steps, update $\theta^- \leftarrow \theta$, $\phi^- \leftarrow \phi$
    end if
end while