Exercise 9.1 (Single to multi-user). (10 points)

\(\ell\)-EUF-CMA Game.

Input: \(\kappa, L\).
Output: ACCEPT or REJECT.

1. For \(P \in L\) do
   \((K_P, k_P) \leftarrow \text{SIG.Keygen}(1^\kappa)\).
2. Invoke the player \(P\) with input
   \((\text{O}_\text{Sig}^{\ell}\text{-EUF-CMA}, \text{O}_\text{Corrupt}^{\ell}\text{-EUF-CMA}, K)\)
   to obtain a party \(P' \in L\) and a message signature pair
   \((m', s')\).
3. If \(\text{O}_\text{Corrupt}^{\ell}\text{-EUF-CMA}\) was called on input \(P'\) then
   Return REJECT.
4. If \(((P', m'), s')\) is the input output pair of a call
   to the oracle \(\text{O}_\text{Sig}^{\ell}\text{-EUF-CMA}\)
   then Return REJECT.
5. Return \(\text{SIG.Vfy}(K_{P'}, m', s')\).

Prove:

Theorem. Any \((t, \varepsilon)\)-EUF-CMA secure signature scheme is also
\((t - t_R, \ell \cdot \varepsilon)\)-\(\ell\)-EUF-CMA secure, where \(t_R \in O(\ell)\) is the overhead runtime
of the reduction.

Hint: Apply Game-Hopping. In some hop, you have to specify a reduction to
the EUF-CMA game and prove that it works.