1. (Puncturable PRFs) Puncturable PRFs are PRFs for which a key can be given out such that, it allows evaluation of the PRF on all inputs, except for one designated input. A puncturable pseudo-random function $F$ is given by a triple of efficient algorithms ($\text{Key}_F, \text{Puncture}_F$, and $\text{Eval}_F$), satisfying the following conditions:

- **Functionality preserved under puncturing**: For every $x^*, x \in \{0, 1\}^n$ such that $x^* \neq x$, we have that:
  \[
  \Pr[\text{Eval}_F(K, x) = \text{Eval}_F(K, x^*) : K \leftarrow \text{Key}_F(1^n), K_{x^*} = \text{Puncture}_F(K, x^*)] = 1
  \]

- **Pseudorandom at the punctured point**: For every $x^* \in \{0, 1\}^n$ we have that for every poly-size adversary $A$ we have that:
  \[
  |\Pr[A(K_{x^*}, \text{Eval}_F(K, x^*)) = 1] - \Pr[A(K_{x^*}, \text{Eval}_F(K, U_n)) = 1]| = \text{negl}(n)
  \]
  where $K \leftarrow \text{Key}_F(1^n)$ and $K_{x^*} = \text{Puncture}_F(K, x^*)$. $U_n$ denotes the uniform distribution over $n$ bits.

Prove that: If one-way functions exist, then there exists a puncturable PRF family that maps $n$ bits to $n$ bits.

**Hint**: The GGM tree-based construction of PRFs from a length doubling pseudorandom generator (discussed in class) can be adapted to construct a puncturable PRF. Also note that $K$ and $K_{x^*}$ need not be the same length.

2. (Proving OR of two statements) Give a statistical zero-knowledge proof system $\Pi = (P, V)$ (with efficient prover) for the following language.

\[
L = \left\{ ((G_0, G_1), (G'_0, G'_1)) \left| G_0 \simeq G_1 \lor G'_0 \simeq G'_1 \right. \right\}
\]

**Caution**: Make sure the verifier doesn’t learn which of the two pairs of graphs is isomorphic.