MODULE *definitions*

LOCAL INSTANCE *Integers*
LOCAL INSTANCE *Sequences*
LOCAL INSTANCE *FiniteSets*
LOCAL INSTANCE *TLC*

Verify a given block header.
$VerifyBlockHeader(proposed\_block,\ tip\_block) \triangleq \text{TRUE}$

Verify a given set of transactions.
$VerifyBlockTransactions(transactions) \triangleq \text{TRUE}$

Verify a zk-SNARK proof for a given transaction.
$VerifyZKProof(proof,\ noteCommitmentRoot,\ nullifierRoot) \triangleq \text{TRUE}$

Generate a sequence of random bytes of length n.
$RandomBytes(n) \triangleq [i \in 1 \,..\, n \mapsto \text{CHOOSE } x \in 0 \,..\, 255 : \text{TRUE}]$

Generate a zk-SNARK proof summarizing the current state
$GenerateZKProof(data) \triangleq RandomBytes(4992)$

Abstract function that computes a new note commitment root given the current state and a set of transactions.
$ComputeNewNoteRoot(oldProof,\ txs) \triangleq RandomBytes(32)$

Abstract function that computes a new nullifier root given the current state and a set of transactions.
$ComputeNewNullifierRoot(oldProof,\ txs) \triangleq RandomBytes(32)$

Create a transaction for a given set of actions.

$$
\begin{array}{l}
OrchardTransaction(actions,\ proof) \triangleq\ [\\
\quad actions \mapsto actions, \\
\quad proof \mapsto proof \\
]
\end{array}
$$