Protocol AgreeOnRandom

A protocol inspired by [BSCG+15] to sample a uniformly random value r among n parties. It requires a commitment scheme (e.g., Scheme **??**), a broadcast channel $\mathcal{F}^{Broadcast}$ and a hash function H (e.g., SHA-256 [Dwo15], a $\mathsf{Transcript}$)

Players: $\mathcal{P}_1, \ldots, \mathcal{P}_i, \ldots, \mathcal{P}_n$
Outputs: r, a random value

$\mathcal{P}_i.\mathsf{Round1}() \dashrightarrow c_i$
1: Sample $r_i \xleftarrow{\$} \mathbb{Z}_{q^*}$ as the random value of $\mathcal{P}_i$.
2: Run $(c_i, w_i) \leftarrow \mathsf{Commit}(r_i)$ to get the commitment c_i and the witness w_i.
3: $\mathcal{F}^{Broadcast}(c_i)$ to distribute all commitments c_i to all parties.

$\mathcal{P}_i.\mathsf{Round2}(\boldsymbol{c} = \{c_1, c_2, ..., c_n\}) \dashrightarrow r_i, w_i$
1: $\mathcal{F}^{Broadcast}(r_i, w_i)$ to reveal r_i and w_i to all parties.

$\mathcal{P}_i.\mathsf{Round3}(\boldsymbol{c} = \{c_1, c_2, ..., c_n\}, \boldsymbol{r} = \{r_1, r_2, ..., r_n\}, \boldsymbol{w} = \{w_1, w_2, ..., w_n\}) \dashrightarrow r$
1: **for** $i \in [n]$ **do**
2: $\quad \mathsf{Open}(r_i, c_i, w_i)$, ABORT if it fails.
3: $r \leftarrow \mathsf{H}(\boldsymbol{r})$
return r

References

[BSCG+15] Eli Ben-Sasson, Alessandro Chiesa, Matthew Green, Eran Tromer, and Madars Virza. Secure sampling of public parameters for succinct zero knowledge proofs. In *2015 IEEE Symposium on Security and Privacy*, pages 287–304. IEEE, 2015.

[Dwo15] Morris Dworkin. Sha-3 standard: Permutation-based hash and extendable-output functions, 2015-08-04 2015.