Week 1 assignment

email: riccardo@fiveelementslabs.com - Discord handle: Reekee - 9483

github repo: https://github.com/RiccardoGalbusera/zk-university

Part 1:

1. Groth16 and PLONK
2. SNARKs require a trusted setup in the sense that users rely on the fact that the setup was done correctly and the keys generated in this process are destroyed. If this wasn't done, someone with access to those keys could sign false transactions.
3. Other differences between SNARKs and STARKs are that:
    - Since STARKs rely on hash functions, they are quantum resistant, unlike SNARKs which are not
    - STARKs have a higher proof size, which makes them more expensive to run. For now, SNARKs are the more gas efficient solution of the two

Part 2:

Question 2:

1. It checks if the output is the result of the multiplication of the two inputs.
2. A powers of tau ceremony is a part of the process to build a trusted environment for zk-SNARKs. This ceremony it's the first step in the generation of the CRS (common reference string), a public parameter which is used to prove and verify. It's important for zk-SNARKs because the CRS is generated by the participants of the ceremony who provide randomness to the procedure, and this ceremony ensures that participants aren't acting maliciously.
3. Phase 1 of a trusted setup randomly produces some parameters by having the participants to the ceremony provide randomness.In Phase 2, those parameters are converted to the CRS.

Question 3:

1.
2. The multiplication a*b*c is a cubic constraint, and circom allows only for at most quadratic constraints

Question 4:

1.
2. Using PLONK didn't require the call to contribute to the key creation. Regarding running time of tests, Groth16 testing is faster (2371ms vs. 3255ms of PLONK)

Question 5:

```
> test
> node scripts/bump-solidity.js && npx hardhat test



  HelloWorld
1x2 = 2
    ✓ Should return true for correct proof (3808ms)
    ✓ Should return false for invalid proof (337ms)

  Multiplier3 with Groth16
1x2 = 2
1x2x4 = 8
    ✓ Should return true for correct proof (2371ms)
    ✓ Should return false for invalid proof

  Multiplier3 with PLONK
1x2 = 2
1x2x4 = 8
    ✓ Should return true for correct proof (3255ms)
    ✓ Should return false for invalid proof
```

Part 3:

Question 1:

1. The 32 in line 9 stands for the number of bits we expect for the two inputs, i.e. we expect the inputs of LessThan to be in 32 bits.
2. The LessThan circuit returns either 0 or 1: 1 if the first input is lower than the second, 0 otherwise.

Question 2:

1.
2. snarkJS: circuit too big for this power of tau ceremony. 98236 > 2**16: the power of tau we downloaded (16 bits) does not support the current circuit. To fix it I downloaded the power of tau for 17 bits.
3.

4. Benefits for this implementation over the brute force method is less computation, which results in lower run time and a better gas optimization