Algebric Circuit to R1CS

Table of Contents

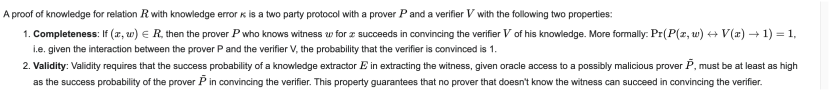
Definitions

Succinct

proof π is succinct if:

1. $|\pi| = \text{poly}(\lambda, \log|\mathcal{C}|)$ and
2. the verification time is $\text{poly}(\lambda, |x|, \log|\mathcal{C}|)$.

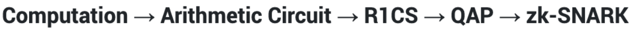
Proof of knowledge



A proof of knowledge for relation R with knowledge error κ is a two party protocol with a prover P and a verifier V with the following two properties:

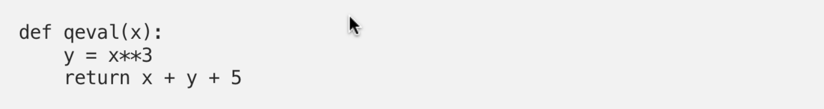
1. **Completeness**: If $(x, w) \in R$, then the prover P who knows witness w for x succeeds in convincing the verifier V of his knowledge. More formally: $\Pr(P(x, w) \leftrightarrow V(x) \rightarrow 1) = 1$, i.e. given the interaction between the prover P and the verifier V, the probability that the verifier is convinced is 1.
2. **Validity**: Validity requires that the success probability of a knowledge extractor E in extracting the witness, given oracle access to a possibly malicious prover $\tilde{P}$, must be at least as high as the success probability of the prover $\tilde{P}$ in convincing the verifier. This property guarantees that no prover that doesn't know the witness can succeed in convincing the verifier.

Construction of SNARKS



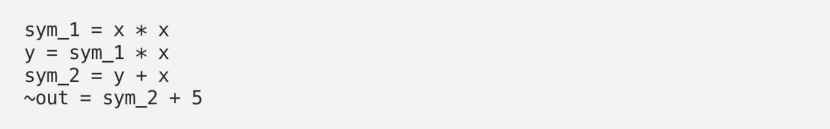
Computation → Arithmetic Circuit → R1CS → QAP → zk-SNARK

Arithmetic circuit to R1CS



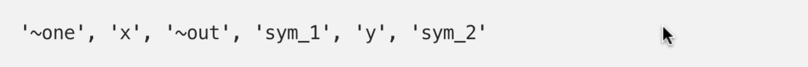
```
def qeval(x):
    y = x**3
    return x + y + 5
```

Arithmetic circuit to R1CS



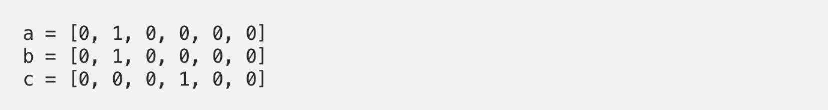
```
sym_1 = x * x
y = sym_1 * x
sym_2 = y + x
~out = sym_2 + 5
```

Arithmetic circuit to R1CS



```
'~one', 'x', '~out', 'sym_1', 'y', 'sym_2'
```

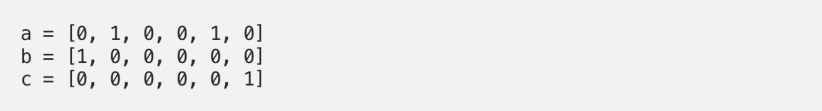
Arithmetic circuit to R1CS



```
a = [0, 1, 0, 0, 0, 0]
b = [0, 1, 0, 0, 0, 0]
c = [0, 0, 0, 1, 0, 0]
```

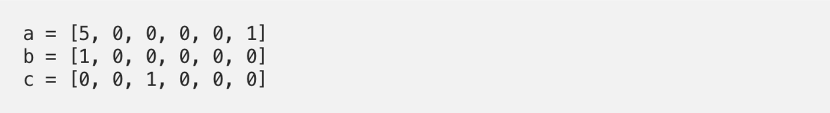
$$s.c = s.a * s.b \implies s = [0, 3, 0, 9, 0, 0]$$

Arithmetic circuit to R1CS



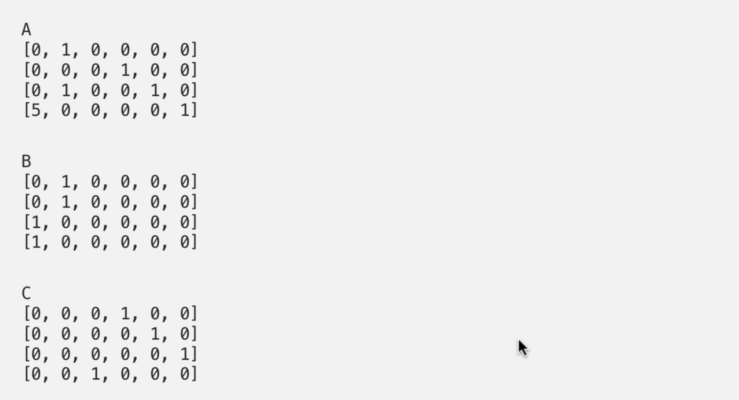
```
a = [0, 1, 0, 0, 1, 0]
b = [1, 0, 0, 0, 0, 0]
c = [0, 0, 0, 0, 0, 1]
```

Arithmetic circuit to R1CS



```
a = [5, 0, 0, 0, 0, 1]
b = [1, 0, 0, 0, 0, 0]
c = [0, 0, 1, 0, 0, 0]
```

Arithmetic circuit to R1CS



```
A
[0, 1, 0, 0, 0, 0]
[0, 0, 0, 1, 0, 0]
[0, 1, 0, 0, 1, 0]
[5, 0, 0, 0, 0, 1]


B
[0, 1, 0, 0, 0, 0]
[0, 1, 0, 0, 0, 0]
[1, 0, 0, 0, 0, 0]
[1, 0, 0, 0, 0, 0]


C
[0, 0, 0, 1, 0, 0]
[0, 0, 0, 0, 1, 0]
[0, 0, 0, 0, 0, 1]
[0, 0, 1, 0, 0, 0]
```

$$\implies (A.s) * (B.s) = C.s$$