```
contract Proxy {

bytes32 private constant implementationPos =
            keccak256("proxy.implementation");


function setAddress(address _implementation) {
  assembly {
    sstore(implementationPos, _implementation)
  }
}


function getAddress() returns(address _implementation) {
  assembly {
    _implementation := sload(implementationPos)
  }
}


function () payable public {
  … delegatecall() …
}


}
```
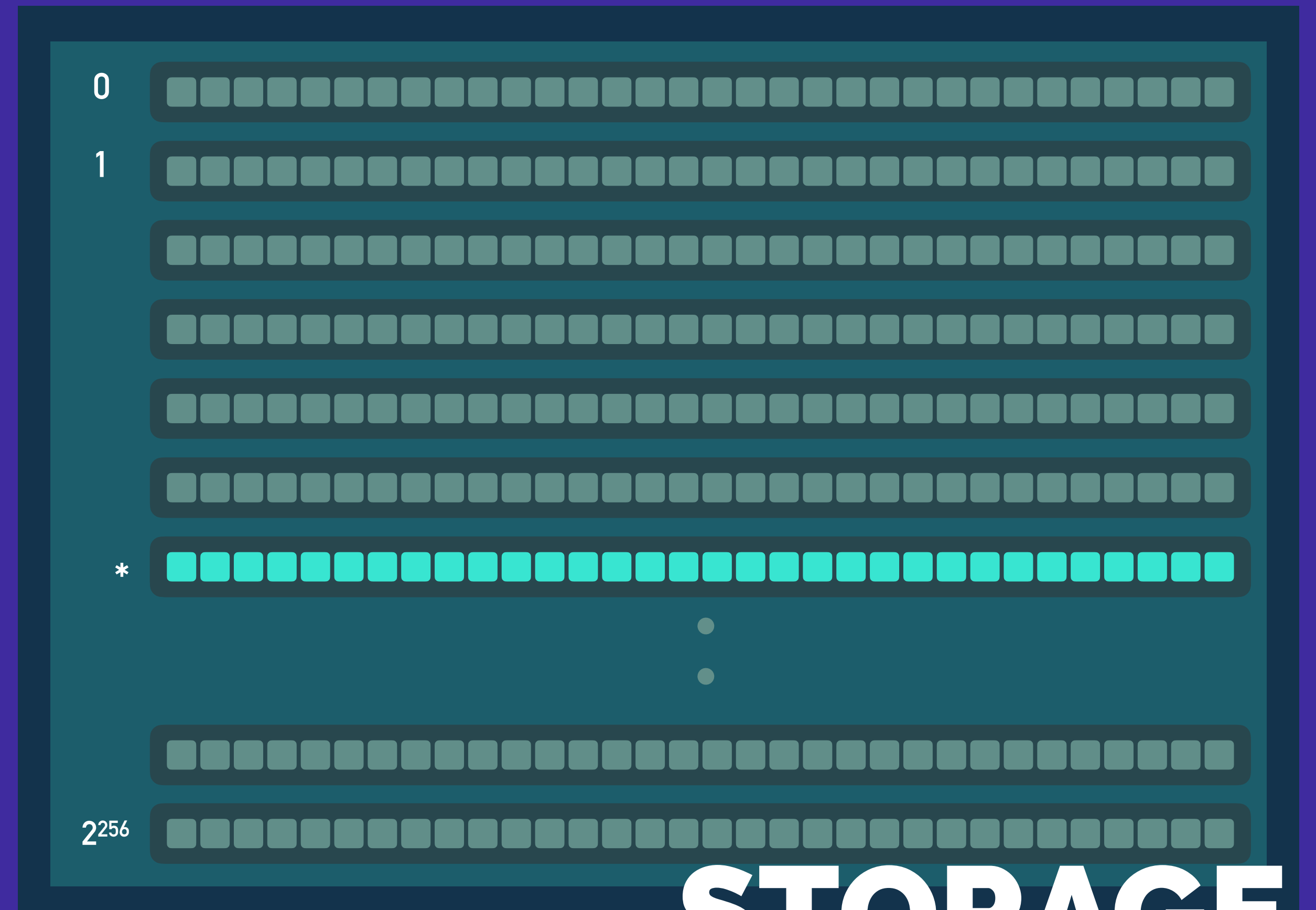
Proxy

STORAGE

0
1

*

2^{256}

```
contract Proxy {

bytes32 private constant implementationPos =
          keccak256("proxy.implementation");


function setAddress(address _implementation) {
  assembly {
    sstore(implementationPos, _implementation)
  }
}


function getAddress() returns(address _implementation) {
  assembly {
    _implementation := sload(implementationPos)
  }
}


function () payable public {
  … delegatecall() …
}


}
```

Proxy

Poll