THE FESTIVAL OF RESEARCH
HOW NOT TO LOSE MILLIONS
SECURING BLOCKCHAINS

Bas Spitters – Associate Professor Computer Science

Thanks: My group
blockchainacademy.dk
VIDEO

Video from blockchainacademy.dk
BLOCKCHAINS

Bulletin boards to post information on (fx. transactions)
Technology underlying bitcoin
Can be used to replace many banking functions

Blockchains (should?) provide trust

1. Decentralization
2. Security
3. Understanding through:
   Open implementation
   Scientific reviewing
   Mathematical proofs
SMART CONTRACTS

Small computer programs that can encode legal contracts

Coins
Escrow
Non-fungible tokens (NFT), art
Crowd Funding
Funding bodies/Governance
Decentralized finance (DeFi)
Small bugs have cost billions of kroners. (DAO, bZx, ...)

DAO distributed funding agency lost millions

bZx token exchange
  wrong order of two lines of code cost 8M$ (BAN blog)

```solidity
function transferFrom(address sender, address receiver, uint256 nrTokens) {
  // 1: uint256 balanceSender = balances[sender]
  // 2: uint256 balanceReceiver = balances[receiver]
  // 3: balances[sender] = balanceSender - nrTokens
  // 4: balances[receiver] = balanceReceiver + nrTokens
  // ...
SOURCES OF BUGS

1. Experimental languages with confusing meaning
2. Errors in translation (compilers)
3. Intricate interaction between contracts
4. Properties of the program are not specified
   "No coins are lost"
AVOIDING BUGS

Use programming languages with a clear meaning

Unit testing
Property testing
Proving
**EXAMPLE**

Natural numbers: 0,1,2,...

Function Sum that sums all numbers 0..n
Sum(n) = 0 + ... + n

Fun Sum (n:nat) :=
\[
\begin{align*}
0 & \Rightarrow 0 \\
\text{m+1} & \Rightarrow \text{Sum}(m) + (m+1) \quad \text{(recursion)}
\end{align*}
\]
EXAMPLE

Sum: 0, 1, 2, 3, 4, ...
0, 1, 3, 6, 10, ...

Looks like $\text{Sum}(n) = \frac{n(n+1)}{2}$

Unit test: works for 5, 12 and 37

Property test: works for a million random numbers

Proving: mathematical proof that it holds for all numbers
By induction or picture
PROVING WITH THE COMPUTER

Computer Proof assistants can (help to) prove such theorems

Used for advanced mathematics (four color, Kepler conjecture)

Advanced computer programs:

Verified compilers (for smart contracts, ...)
Cryptographic implementations
PROOFS ABOUT BLOCKCHAINS

NAKAMOTO CONSENSUS safety and liveness

Proofs about paradigmatic smart contracts
automatic testing and correct translations

High assurance cryptographic implementations
(Correct and Secure)

See CoBRA website
CONCLUSION

- Blockchains and smart contracts can automate traditional banking tasks
- Bugs are costly
- Logic and mathematics with the computer help to avoid many bugs

blockchainacademy.dk