Listen to \( \frac{n+1}{2} \) parties saying "\( p \rightarrow D \)".

\( t \) corrupted parties

Listen to \( >t+1 \) parties saying "\( p \rightarrow D \)".

1980: Byzantine Generals Problem.

2008: Bitcoin by Satoshi Nakamoto.

Nakamoto consensus: "longest chain wins".

Cryptographic hash function:

\[
\begin{array}{c}
\text{oracle} \\
H \\
\uparrow \\
x \\
\downarrow \\
y = H(x) \\
\end{array}
\]

If \( x \) was queried, return \( H(x) = y \).

Toss random coins (256 coins)

\( \downarrow \)

Remember \( (H(x) = y) \).

Output \( y \).

\( \Rightarrow \) deterministic.
L \xrightarrow{x}: arbitrarily large input
y: fixed size (256 bits) 512 bit.

\[ \exists x, x' \text{ s.t. } H(x) = H(x') \]

Collision resistance: It is infeasible to find \( x \) & \( x' \) s.t. \( H(x) = H(x') \)

\[ \downarrow \]

Second pre-image resistance: Given \( x \), it is infeasible to find \( x' \), etc.

\[ \downarrow \]

Pre-image resistance: Given \( H(x) \) it is hard to find \( x \).

Immutability.

Longest chain wins rule.
Each party maintains many different chains C.
Protocol proceeds in rounds:
1. In each round, we elect a unique random leader L.
2. If L has at least one chain in its storage, it picks the longest chain C ∈ G, creates a new empty B and attaches C ← B.
   O.W. if C is empty, start a new chain with a block containing $P \rightarrow D$ or $P \rightarrow S$.

Announce chain to all other parties. (If some party receives any information, send it to everyone)

Commit: when the longest chain is $k$ blocks deep; commit the first block.

Scenario I: All parties are honest.

```
P→P
  |   |
  |   |
  |   |
  |   |
  |   |
  |   |
  |   |  k
P→S
```


Scenario 2: Some malicious parties (50%).

Alternative exec:

Scenario 3: \# malicious parties $\geq 90\%$.

Scenario 4: \# malicious parties = 49%.

# honest > # malicious.
$\text{In (worst case)} \approx \frac{1}{12}$

when some chain is length $k$,

$\Pr[\text{Private chain attack succeeds}] \leq \exp(-k)$.

$30Y. > 0.2$