



School: Campus:
Academic Year: Subject Name: Subject Code:
Semester: Program: Branch: Specialization:
Date:

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment :

* Coding Phase: Pseudo Code / Flow Chart / Algorithm

ALGORITHM:

1.Start

2. Collect transaction data to form a block.

3. Set a **target difficulty** (a limit for the hash value).

4. Initialize a **nonce = 0**.

5. Combine block data + nonce → apply **hash function (SHA-256)**.

6. Check if the **hash < target difficulty**:

- If **yes**, block is successfully mined.
- If **no**, increment nonce and repeat Step 5.

7. Broadcast the mined block to the network.

8. Other nodes verify the hash and add the block to the blockchain.

9.Stop

* Softwares used

Bitcoin Core – the original full-node and mining software for Bitcoin.

CGMiner – popular open-source mining software supporting ASIC and GPU mining.

BFGMiner – modular miner for ASIC/FPGA devices, allows remote monitoring.

NiceHash Miner – beginner-friendly software that automatically selects profitable algorithms.

* Testing Phase: Compilation of Code (error detection)

Open the Proof of Work Simulator On your brave browserThe page will load a visual simulator with multiple blocks.



COURSE OVERVIEW TOOL OVERVIEW ▾ BLOG FAQ LOGIN 🔍

Proof of Work Simulator

Published by [Mario Oettler](#) on 28. May 2021

Last Updated on 12. August 2024 by Martin Schuster

Proof of Work Simulator

Block Nr #1	previous hash:
Nonce: 77318	00000000000000000000000000000000
Data:	Hash: 00460059210f0f0654fcb6fe685e

MINE

Understand the Layout You'll see blocks labeled Block #1, Block #2, etc.

Each block has: Data (text field), Nonce (number), Previous Hash (link to previous block), Hash (current block hash), Mine button.

Mine the First Block Click the "Mine" button on Block #1. The simulator will start calculating a valid nonce. Once the hash of the block starts with required zeroes (like 00...), the block turns green (valid). Now Block #1 is mined successfully.

Proof of Work Simulator

Proof of Work Simulator

Block Nr #1	previous hash:
Nonce: 89161	00000000000000000000000000000000
Data: amit kumar	Hash: 002017b26f43fe8d9d8aad2002c1

MINE

Mine the Next Block (Block #2), Block #2 takes the hash of Block #1 as its "Previous Hash". Click the "Mine" button on Block #2. Again, the simulator finds a valid nonce and turns the block green once it's valid.

* Testing Phase: Compilation of Code (error detection)

Block Nr #2	
Nonce:	previous hash:
Data:	Hash:
Block Nr #3	
Nonce:	previous hash:
Data:	Hash:

Continue Mining All Blocks Repeat the process for Block #3 and Block #4. Each block is dependent on the hash of the previous block.

Modify the Block Data Now try changing the Data field in Block #1. You'll see that the hash changes, and Block #1 and all blocks after it turn red. This shows the chain is broken due to tampering—this is how blockchain ensures immutability.

Block Nr #1	previous hash:
Nonce:	00000000000000000000000000000000
89161	
Data:	Hash:
amit kumar	002017b26f43fe8d9d8aad2002c1
MINE	

Block Nr #1	previous hash:
Nonce:	00000000000000000000000000000000
68688	
Data:	Hash:
newr	0017632569e4d71612ca3c8bb44f
MINE	
Block Nr #2	previous hash:
Nonce:	00fc944070f442361db90260990c
20357	
Data:	Hash:
newlndndnd	00015f2ddf861daec1037a2560f0
MINE	

* Implementation Phase: Final Output (no error)

Now you see i successfully completed the mining of all the blocks

Proof of Work Simulator

Published by [Mario Oettler](#) on 28. May 2021

Last Updated on 12. August 2024 by [Martin Schuster](#)

Block Nr #1

previous hash: 00000000000000000000000000000000

Nonce: 89161

Data: Hash: amit kumar 002017b26f43fe8d9d8aad2002c1

MINE

Block Nr #2

previous hash: 002017b26f43fe8d9d8aad2002c1

Nonce: 34660

Data: Hash: quick mine 004e50f496349614c28cba8fa992

MINE

Block Nr #3

previous hash: 004e50f496349614c28cba8fa992

Nonce: 88449

Data: Hash: not available 00f919f7bde3e47689f6e9a2a50b

MINE

Block Nr #4

previous hash: 00f919f7bde3e47689f6e9a2a50b

Nonce: 29338

Data: Hash: mining 0078ccfac17ed22d37c408352cd7

MINE

CLEAR

* Observations

- >Mining requires **high computational power** to solve complex hash puzzles.
- >**Nonce value** changes continuously until the correct hash is found.
- >The **hash output** must be below the target difficulty.
- >**More powerful hardware** increases the chance of mining a block.
- >Mining consumes **a large amount of electricity**.
- >Once a valid block is mined, it is **broadcast and verified** by other nodes.
- >The successful miner receives a **block reward** (cryptocurrency + transaction fees).

ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10		
Planning and Execution/ Practical Simulation/ Programming	10		
Result and Interpretation	10		
Record of Applied and Action Learning	10		
Viva	10		
Total	50		

Signature of the Student:

Name :

Regn. No. :

***As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.**

Signature of the Faculty: