PROOF OF WORK - II

BLOCKCHAIN DEVELOPMENT

Time: 60 mins

Introduction

In this class, the student/s will be able to use nonce value to generate the hash with the given difficulty level.

Introduce and Recall Commands

- time()

  Time function returns the number of seconds passed since epoch.

- __init__()

  Python __init__() constructor is an constructor in object oriented approach. This function is called every time an object is created from a class.

- self

  A default parameter, named 'self' is always passed in its argument. This self represents the object of the class itself.

Vocabulary

- **Nonce:** A nonce is a value that can be used only once. It is an artificially generated number used as counter during the process of mining.

Learning Objectives

Student/s should be able to:

- ***Recall*** how a problem is solved using a nonce.
- ***Explain*** how a problem is solved using a nonce to match the difficulty level.
- ***Demonstrate*** how a nonce value was found to match the difficulty level.

Activities

1. **Class Narrative:** *(2 mins)*

   - Brief the student/s that they will learn to solve the complex math problem to generate the hash to validate the block.

2. **Concept Introduction Activity:** *(5 mins)*

- Let the student/s play the explore-activity to observe .
- Explain the need of hashing in blockchain and introduce Secure hashing for data.
- Using the slides, explain that the student/s will learn:
    - to generate sequence of nonce
    - to limit the nonce
    - To validate proof of work

3. **Activity 1: GENERATE SEQUENTIAL NONCE** *(14 mins)*

    Teacher Activity: *(7 mins)*

- Explain why nonce value is used instead of random numbers/string.
- Demonstrate how to add the nonce value to the block by passing it through the functions.

    Student Activity: *(7 mins)*

- Guide the student/s to add nonce value to the block and update the functions to use nonce value to generate the hash..

4. **Activity 2: LIMIT THE NONCE** *(12 mins)*

    Teacher Activity: *(6 mins)* .

- Explain why limiting the nonce is necessary and how to limit the nonce.
- Demonstrate how to limit the nonce and add a condition to check for the nonce limit.

    Student Activity: *(6 mins)*

- Guide the student/s to limit the nonce and add a condition to check for the nonce limit.

5. **Activity 3: VALIDATE PROOF OF WORK** *(12 mins)*

    Teacher Activity: *(6 mins)*

- Explain that the nonce will be used to validate the block hash.
- Explain that the hash can be validated only when the hash matches the difficulty level.

- Explain how timestamp can be used to create new hash and reuse the nonce value as nonce are limited.
- Demonstrate how to calculate the hash to match the difficulty level and use the timestamp to reuse the nonce values.

  Student Activity: *(6 mins)*

- Guide the students to calculate the hash to match the difficulty level and use the timestamp to reuse the nonce values.

6. **Test and Summarize the class learnings:** *(5 mins)*
   - Check for understanding through quizzes and summarize learning after respective missions.
   - Summarize the overall class learning towards the end of the class.
7. **Additional activities:**
   - Encourage the student/s to use the unique nonce for each block.
   - Encourage the student/s to add a time limit to mine the block. Also decrease the limit of nonce and increase the difficulty level.
8. **State the Next Class Objective:** *(1 min)*
   - In the next class, student/s will learn to create a mining pool, reward the miners after solving the problem to validate the block.

U.S. Standards:

CSTA: 2-AP-11, 2-AP-12, 2-AP-13, 2-AP-14, 2-AP-19

Links Table		
Activity	**Activity Name**	**Link**
Class Presentation	PROOF OF WORK	https://s3-whjr-curriculum-uploads.whjr.online/1cf64c47-4900-4060-838e-1c6e376e77b1.html
Explore Activity	PROOF OF WORK	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS
Teacher Activity 1	GENERATE SEQUENTIAL NONCE	https://github.com/Tynker-Blockchain/TNK-M12-C93-TAS-BP

Teacher Reference: Teacher Activity 1 Solution	GENERATE SEQUENTIAL NONCE	https://github.com/Tynker-Blockchain/TNK-M12-C93-TAS
Student Activity 1	GENERATE SEQUENTIAL NONCE	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS-BP
Teacher Reference: Student Activity 1 Solution	GENERATE SEQUENTIAL NONCE	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS
Teacher Activity 2	LIMIT THE NONCE	https://github.com/Tynker-Blockchain/TNK-M12-C93-TAS-BP
Teacher Reference: Teacher Activity 2 Solution	LIMIT THE NONCE	https://github.com/Tynker-Blockchain/TNK-M12-C93-TAS
Student Activity 2	LIMIT THE NONCE	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS-BP
Teacher Reference: Student Activity 2 Solution	LIMIT THE NONCE	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS
Teacher Activity 3	VALIDATE PROOF OF WORK	https://github.com/Tynker-Blockchain/TNK-M12-C93-TAS-BP
Teacher Reference: Teacher Activity 3 Solution	VALIDATE PROOF OF WORK	https://github.com/Tynker-Blockchain/TNK-M12-C93-TAS
Student Activity 3	VALIDATE PROOF OF WORK	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS-BP
Teacher Reference: Student Activity 3 Solution	VALIDATE PROOF OF WORK	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS
Student's Additional Activity 1	UNIQUE NONCE	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS-BP
Teacher Reference: Student's Additional Activity 1 Solution	UNIQUE NONCE	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS
Student's Additional Activity 2	LIMIT THE NONCE VALUE AND INCREASE THE DIFFICULTY LEVEL	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS-BP
Teacher Reference: Student's Additional Activity 2 Solution	LIMIT THE NONCE VALUE AND INCREASE THE DIFFICULTY LEVEL	https://github.com/Tynker-Blockchain/TNK-M12-C93-SAS
Post Class Project	VALIDATE PROOF OF WORK	https://github.com/Tynker-Blockchain/TNK-M12-C93-PCP-BP
Teacher Reference: Post Class Project Solution	VALIDATE PROOF OF WORK	https://github.com/Tynker-Blockchain/TNK-M12-C93-PCP