Sipna College of Engineering & Technology, Amravati.
Department of Computer Science & Engineering

Branch :- Computer Sci. & Engg. **Class :- Final Year**
Subject :-Block Chain Fundamentals Lab manual **Sem :- VII**

Teacher Manual

PRACTICAL NO 5

AIM: Create a simple BlockChain

S/W REQUIRED: Phython

Block Chain

A block chain is a distributed database or ledger that is shared among the nodes of a computer network. As a database, a block chain stores information electronically in digital format. Block chains are best known for their crucial role in cryptocurrency systems, such as Bitcoin, for maintaining a secure and decentralized record of transactions. The innovation with a block chain is that it guarantees the fidelity and security of a record of data and generates trust without the need for a trusted third party.

Implementation:

```
from hashlib import sha256
import time

class block:
        def __init__ (self, timestamp, data, previousHash = ' '):
                self.timestamp = timestamp
                self.data = data
                self.previousHash = previousHash
                self.hash = self.calculateHash()

        def calculateHash(self):
                return sha256((str(self.timestamp) + str(self.data) +
str(self.previousHash)).encode()).hexdigest()


class blockchain:
        def __init__(self):
                self.chain = [self.createGenesis()]

        def createGenesis(self):
                return block(time.ctime(), "genesisBlock", "00000")

        def mineBlock(self, data):
                node = block(time.ctime(), data, self.chain[-1].hash)
                # mining a new block to the blockchain
                self.chain.append(node)

        def printBlockchain(self):
                for i in range(len(self.chain)):
                        print("\n-----Block ", i ,"---------\n timestamp = "\
                                , self.chain[i].timestamp,"\n data = ", \
```

```
                                        self.chain[i].data, "\n previousHash = ",\
                                         self.chain[i].previousHash,"\n hash = ", \
                                            self.chain[i].hash)


CEVcoin = blockchain()

data = input()

# sending data to get mined
print("\n\n ----> Mining New Block -->")
CEVcoin.mineBlock(data)

print("\n\n ----> New Block mined successfully --> ")

CEVcoin.printBlockchain()
```

Output:

```
----> New Block mined successfully -->

----Block  0 ---------
timestamp =  Thu Aug 18 13:42:34 2022
data =  genesisBlock
previousHash =  00000
hash =  d35695bf15937b88e66f362bff9ba94ea329cce7a1b2b690127e38fd18b766d9

----Block  1 ---------
timestamp =  Thu Aug 18 13:44:22 2022
data =
previousHash =  d35695bf15937b88e66f362bff9ba94ea329cce7a1b2b690127e38fd18b766d9
hash =  546e3f89d05247109797e377309fedf1b5e89c2917a769e07eff8fe544a949cd
S C:\Users\PC-1059\hello>
S C:\Users\PC-1059\hello>
```

CONCLUSION: Thus we have created a block chain .