Foundation of Blockchain (4361603) - Summer 2025 Gujarati Solution

Milav Dabgar

May 14, 2025

પ્રશ્ન 1(અ) [3 ગુણ]

બ્લોકચેનમાં Private key અને Public key નો તફાવત આપો.

બાબત	Private Key	Public Key
હેતુ	Transaction sign કરવા માટે	Verification માટે ઉપયોગ
શેરિંગ	ગુપ્ત રાખવી જોઈએ	બધાને આપી શકાય
કામ	Data decrypt કરે, signature બનાવે	Data encrypt કરે, signature verify કરે
માલિકી	ફક્ત માલિક જ જાણે	બધા access કરી શકે

- **Private Key**: ગુપ્ત mathematical code જે ownership સાબિત કરે
- **Public Key**: ખુલ્લું address જેથી બીજા transaction મોકલી શકે
- **સુરક્ષા**: Private key ગુમાવવી = પૈસા હંમેશ માટે ગુમાવવા

મેમરી ટ્રીક

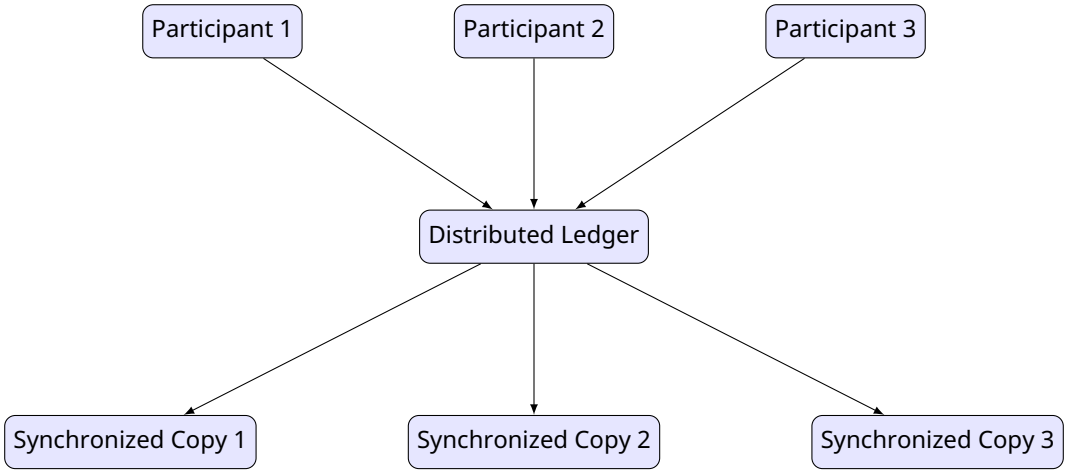
Private છે Personal, Public છે Posted

પ્રશ્ન 1(બ) [4 ગુણ]

Distributed Ledger ને વિગતવાર સમજાવો.

Distributed Ledger એ database છે જે ઘણી જગ્યાએ અને ઘણા લોકોમાં વહેંચાયેલું હોય છે.

લક્ષણ	વર્ણન
Decentralized	કોઈ એક control point નથી
Synchronized	બધી copies updated રહે છે
Transparent	બધા participants જોઈ શકે છે
Immutable	સહેલાઈથી બદલાતું નથી



આકૃતિ 1. Distributed Ledger System

- **ફાયદા**: Intermediaries નાબૂદ કરે, trust વધારે, fraud ઓછું
- **કામ**: બધા participants પાસે records ની identical copies હોય

મેમરી ટ્રીક

Distributed = વિભાજિત પણ સમાન

પ્રશ્ન 1(ક) [7 ગુણ]

Blockchain વ્યાખ્યાયિત કરો. Blockchain ની એપ્લિકેશનો અને મર્યાદાઓનાં વર્ણન કરો.

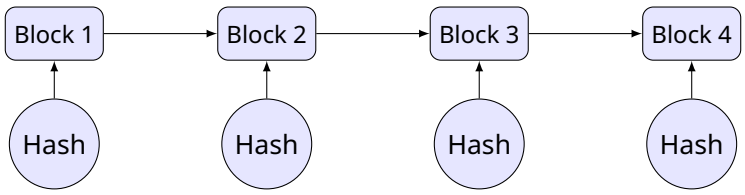
Blockchain વ્યાખ્યા: Transaction records ધરાવતા blocks નો chain જે cryptography વાપરીને જોડાયેલા હોય.

એપ્લિકેશન કોષ્ટક:

ક્ષેત્ર	એપ્લિકેશન	ફાયદો
Finance	Cryptocurrency, payments	ઝડપી, સસ્તી transfers
Healthcare	Patient records	સુરક્ષિત, accessible data
Supply Chain	Product tracking	પારદર્શિતા, authenticity
Real Estate	Property records	Fraud prevention
Voting	Digital elections	પારદર્શી, tamper-proof

મર્યાદાઓ કોષ્ટક:

મર્યાદા	અસર
Scalability	ધીમી transaction processing
Energy Usage	વધુ electricity વપરાશ
Complexity	Users માટે સમજવું મુશ્કેલ
Regulation	કાયદાકીય અસ્પષ્ટતા
Storage	વધતો data size ની સમસ્યા



આકૃતિ 2. Blockchain Architecture

- **સુરક્ષા**: Cryptographic linking થી tampering મુશ્કેલ
- **પારદર્શિતા**: બધા transactions network participants ને દેખાય

મેમરી ટ્રીક

Blocks Chained = Blockchain, Apps ઘણી = Limits ઘણી

OR

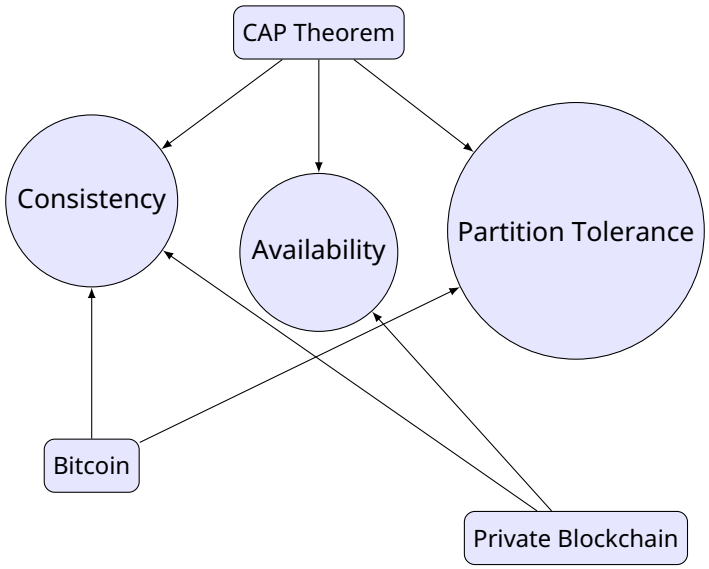
પ્રશ્ન 1(ક) [7 ગુણ]

ટૂંકી નોંધ લખો: બ્લોકચેનમાં CAP Theorem

CAP Theorem કહે છે કે distributed systems એ 3 properties માંથી માત્ર 2 જ simultaneously guarantee કરી શકે.

CAP Components કોષ્ટક:

Property	વર્ણન	ઉદાહરણ
Consistency	બધા nodes પાસે same data	બર્યાને જગ્યાએ same balance દેખાય
Availability	System હંમેશા response આપે	Network કદી down ન જાય
Partition Tolerance	Network failures છતાં કામ કરે	Nodes disconnect થયા છતાં function કરે



આકૃતિ 3. CAP Theorem and Blockchain Trade-offs

વાસ્તવિક ઉપયોગ:

Blockchain Type	પસંદ કરે	ત્યાગ કરે
Bitcoin	Consistency + Partition	Availability
Ethereum	Consistency + Partition	Availability
Private Networks	Consistency + Availability	Partition Tolerance

- **અસર**: Blockchain designers એ કયા property sacrifice કરવી તે choose કરવું પડે
- **Trade-off**: Distributed networks માં perfect systems અશક્ય

મેમરી ટ્રીક

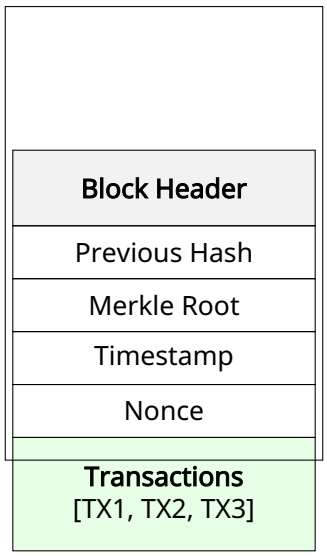
કમ્પ્લીટ સિસ્ટમ શક્ય નથી - 3 માંથી 2 જ પસંદ કરો

પ્રશ્ન 2(અ) [3 ગુણ]

બ્લોકચેનનાં Data Structure સમજાવો.

Blockchain Data Structure transaction data ધરાવતા linked blocks ધાયેલું હોય છે.

Component	હેતુ
Block Header	Metadata રાખે છે
Previous Hash	Previous block સાથે link કરે
Merkle Root	બધા transactions નો summary
Timestamp	Block કયારે બન્યો તેની માહિતી
Transactions	વાસ્તવિક data/transfers



આકૃતિ 4. Structure of a Block

- **Linking**: દરેક block previous block ને hash વાપરીને point કરે
- **Integrity**: એક block બદલાવવાથી આખી chain ટૂટી જાય

મેમરી ટ્રીક

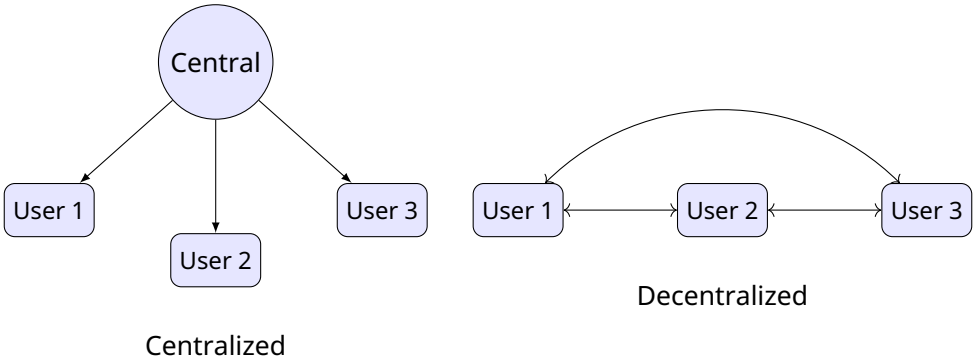
Header હોય છે, Transactions વાત કરે છે

પ્રશ્ન 2(બ) [4 ગુણ]

Decentralization ના ફાયદા શું છે?

Decentralization ફાયદા:

ફાયદો	સમજૂતી
No Single Point of Failure	એક node fail થયા છતાં network ચાલુ રહે
Censorship Resistance	કોઈ authority transactions block કરી શકે નહિ
Transparency	બધા participants સમાન માહિતી જુએ છે
Reduced Costs	Intermediary fees નાબૂદ થાય
Trust	Central authority પર trust કરવાની જરૂર નથી



આકૃતિ 5. Centralized vs. Decentralized Networks

- **સુરક્ષા**: Multiple copies થી data loss અટકે
- **લોકશાહી**: બધા participants ને સમાન અધિકાર
- **મજબૂતાઈ**: Individual failures સામે system ટકે

મેમરી ટ્રીક

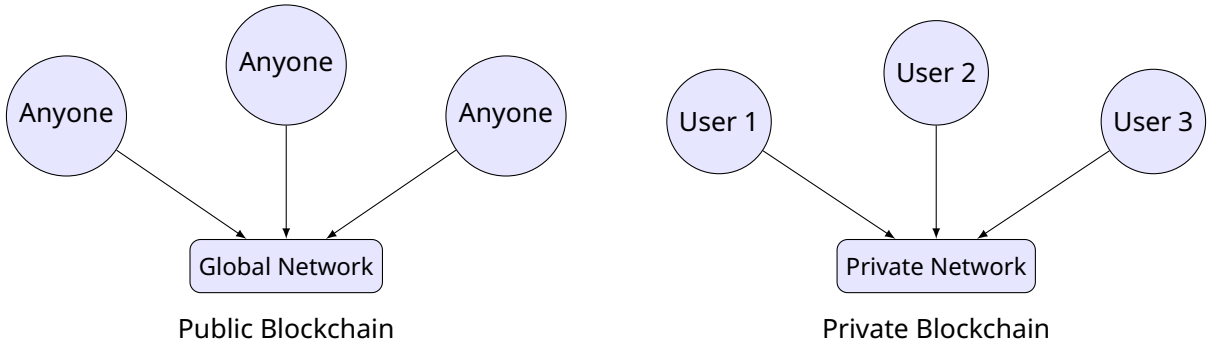
વિકેન્દ્રિત = ટકાઉ, લોકશાહી, પ્રત્યક્ષ

પ્રશ્ન 2(ક) [7 ગુણ]

Public બ્લોકચેન અને Private બ્લોકચેન વચ્ચે તફાવત કરો.

વ્યાપક સરખામણી:

બાબત	Public Blockchain	Private Blockchain
Access	બધા માટે ખુલ્લું	ખાસ users માટે મર્યાદિત
Permission	Permission ની જરૂર નથી	Permission આવશ્યક
Control	Decentralized	Centralized control
Speed	ધીમું (consensus જરૂરી)	ઝડપી (ઓછા validators)
Security	ઊંચી (ઘણા validators)	મધ્યમ (ઓછા validators)
Cost	Transaction fees જરૂરી	ઓછી operational costs
Transparency	સંપૂર્ણ પારદર્શિતા	મર્યાદિત પારદર્શિતા
ઉદાહરણ	Bitcoin, Ethereum	Hyperledger, R3 Corda



આકૃતિ 6. Public vs Private Architecture

- **Trade-offs**: Public વધુ security આપે, Private વધુ control આપે
- **પસંદગી**: Transparency vs. privacy ની જરૂરિયાત પર નિર્ભર

મેમરી ટ્રીક

Public = લોકોનું, Private = મંજૂરીવાળું

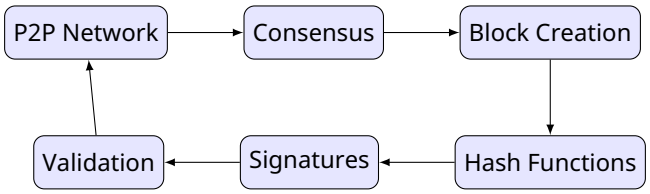
OR

પ્રશ્ન 2(અ) [3 ગુણ]

યોગ્ય આકૃતિ સાથે બ્લોક ચેઇનના Core Components નાં વર્ણન કરો.

મુખ્ય Components:

Component	કામ
Blocks	Transaction data store કરે
Hash Functions	Unique fingerprints બનાવે
Digital Signatures	Transaction authenticity verify કરે
Consensus Mechanism	Valid transactions પર સંમતિ કરે
Peer-to-Peer Network	બધા participants ને connect કરે



આકૃતિ 7. Blockchain Core Components Interaction

- **એકીકરણ**: બધા components મળીને security માટે કામ કરે
- **હેતુ**: દરેક component ખાસ blockchain function serve કરે

મેમરી ટ્રીક

Blocks બનાવે, Hash પકડે, Signatures સુરક્ષિત કરે

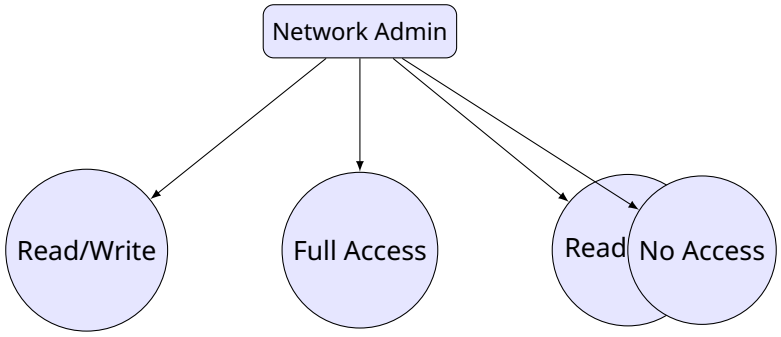
OR

પ્રશ્ન 2(બ) [4 ગુણ]

Permissioned blockchain ને વ્યાખ્યાયિત કરો અને વિગતવાર સમજાવો.

Permissioned Blockchain વ્યાખ્યા: એવી blockchain જેમાં participation માટે network administrators પાસેથી સ્પષ્ટ permission જરૂરી હોય.

લક્ષણ	વર્ણન
Access Control	ફક્ત approved users જ join કરી શકે
Validation Rights	પસંદગીના nodes જ transactions validate કરે
Governance	Central authority network manage કરે
Privacy	Transaction details private હોઈ શકે



આકૃતિ 8. Permission Levels in Permissioned Blockchain

- **ફાયદા**: બહેતર privacy, regulatory compliance, ઝડપી processing
- **ગેરફાયદા**: ઓછું decentralized, administrators પર trust આવશ્યક

મેમરી ટ્રીક

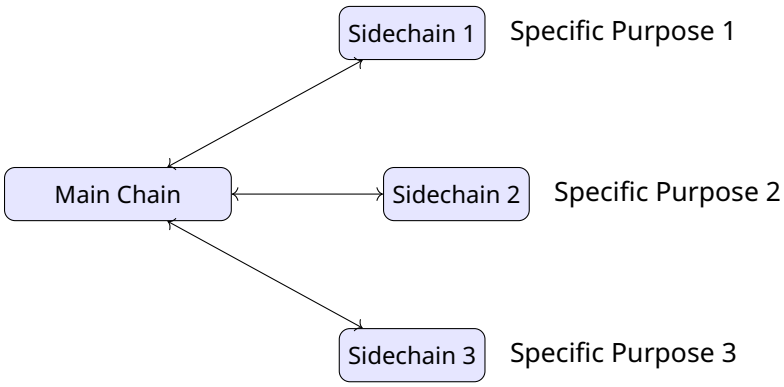
Permission = Participation માટે મંજૂરી

OR

પ્રશ્ન 2(ક) [7 ગુણ]

Sidechain ને સંક્ષિપ્તમાં સમજાવો.

Sidechain વ્યાખ્યા: Main blockchain સાથે connected અલગ blockchain જે chains વચ્ચે asset transfer કરવાની સુવિધા આપે.



આકૃતિ 9. Sidechain Architecture

ફાયદા અને લક્ષણો:

બાબત	ફાયદો
Scalability	Main chain નો load ઘટાડે
Experimentation	નવા features સુરક્ષિત રીતે test કરે
Specialization	ખાસ use cases માટે optimized
Interoperability	અલગ અલગ blockchains ને connect કરે

Transfer Process:

1. **Lock**: Main chain પર assets lock કરાય
2. **Proof**: Cryptographic proof generate કરાય
3. **Release**: Sidechain પર equivalent assets release કરાય
4. **Use**: Sidechain પર assets ઉપયોગ કરાય
5. **Return**: Assets પાછા લાવવા માટે reverse process

વાસ્તવિક ઉદાહરણો:

Sidechain	હેતુ
Lightning Network	ઝડપી Bitcoin payments
Plasma	Ethereum scaling
Liquid	Bitcoin trading

- **સુરક્ષા**: Secure main chain સાથેનું connection જાળવે
- **લવચિકતા**: દરેક sidechain ના અલગ rules હોઈ શકે

મેમરી ટ્રીક

Side સહાય કરે, Main જાળવે

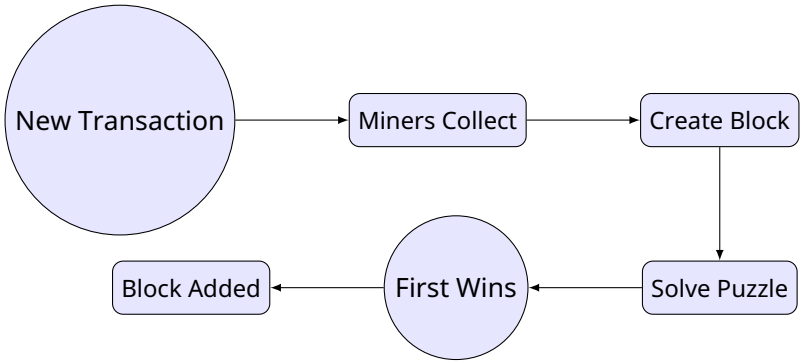
પ્રશ્ન 3(અ) [3 ગુણ]

Consensus Mechanism ને વ્યાખ્યાયિત કરો અને કોઈપણ એકને વિગતવાર સમજાવો.

Consensus Mechanism વ્યાખ્યા: એક protocol જે ખાતરી કરે કે બધા network participants blockchain ની current state પર સંમત હોય.

Proof of Work (PoW) સમજૂતી:

Component	કામ
Mining	જટિલ mathematical puzzles solve કરવું
Competition	Miners વચ્ચે પહેલા solve કરવાની સ્પર્ધા
Verification	Network solution verify કરે
Reward	Winner ને cryptocurrency reward મળે



આકૃતિ 10. Proof of Work Process

- **સુરક્ષા**: Computational work થી tampering મોંઘું બને
- **ઉદાહરણ**: Bitcoin Proof of Work consensus વાપરે

મેમરી ટ્રીક

Consensus = સામાન્ય બુદ્ધિ, Work = જીત

પ્રશ્ન 3(બ) [4 ગુણ]

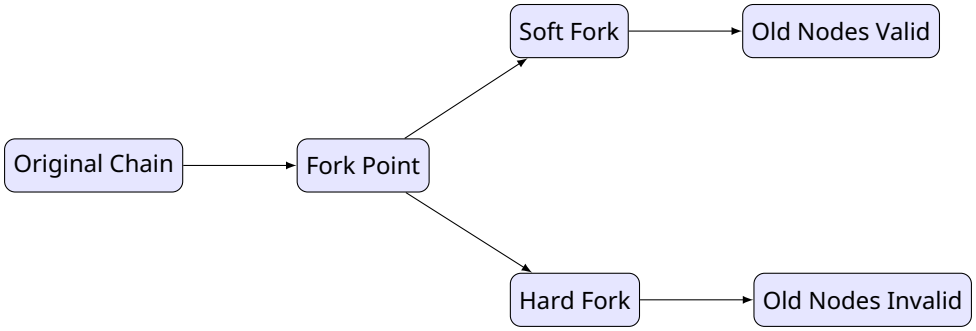
બ્લોકચેનમાં Forking શા માટે જરૂરી છે? બ્લોકચેનમાં વિવિધ પ્રકારના Forks ની યાદી બનાવો.

Forking કેમ જરૂરી:

કારણ	હેતુ
Upgrades	Blockchain માં નવા features add કરવા
Bug Fixes	Security vulnerabilities સુધારવા
Rule Changes	Consensus rules modify કરવા
Community Disagreement	Consensus ન મળે ત્યારે split કરવા

Forks ના પ્રકારો:

Fork Type	વર્ણન	Compatibility
Soft Fork	Rules tight કરે	Backward compatible
Hard Fork	Rules સંપૂર્ણ બદલે	Backward compatible નથી
Accidental Fork	અસ્થાયી split	આપોઆપ resolve થાય
Contentious Fork	Community disagreement	કાયમી split



આકૃતિ 11. Soft vs Hard Fork

- **અસર**: Forks થી નવી cryptocurrencies બની શકે
- **ઉદાહરણો**: Bitcoin Cash (hard fork), Ethereum updates (soft forks)

મેમરી ટ્રીક

Fork = ભવિષ્યના વિકલ્પો, Rules જાળવાય

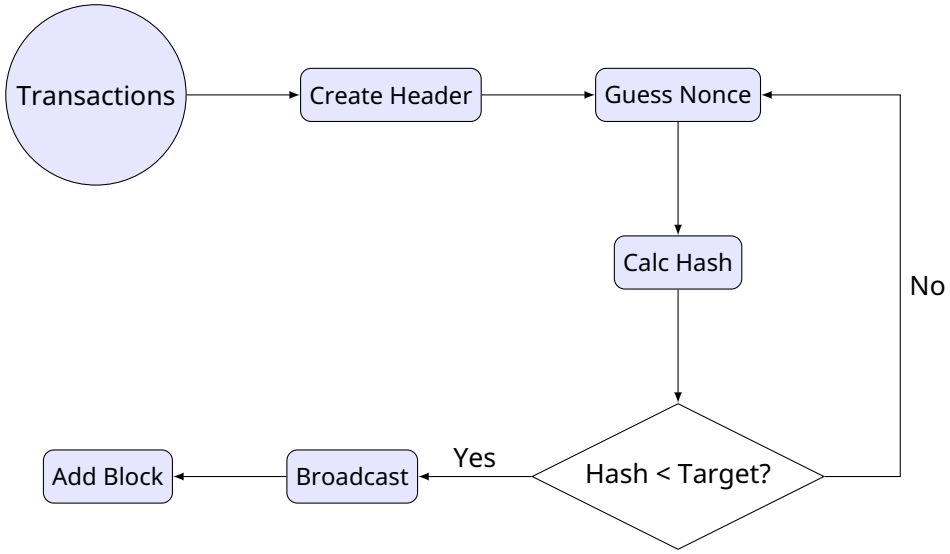
પ્રશ્ન 3(ક) [7 ગુણ]

Bitcoin Mining શું છે? Bitcoin Mining નાં કામકાજ, મુશ્કેલી અને ફાયદાઓ વિશે વિગતવાર જણાવો.

Bitcoin Mining વ્યાખ્યા: Computational puzzles solve કરીને Bitcoin blockchain માં નવા transactions add કરવાની પ્રક્રિયા.

Mining Process:

1. **Collection**: Pending transactions ભેગા કરવા (Mempool માંથી)
2. **Block Creation**: નવો block બનાવવો અને transactions સામેલ કરવા
3. **Puzzle Solving**: સાચો nonce શોધવો (Trial and error)
4. **Verification**: Network solution check કરે અને block validate કરે
5. **Addition**: Chain માં block add કરવો (કાયમી record)
6. **Reward**: Miner ને Bitcoin મળે (હાલમાં 6.25 BTC)



આકૃતિ 12. Bitcoin Mining Workflow

Difficulty Adjustment:

બાબત	પદ્ધતિ
Target Time	દરેક block માટે 10 મિનિટ
Adjustment Period	દરેક 2016 blocks (2 અઠવાડિયા)
Auto-Regulation	Blocks ઝડપી આવે તો વધારે
હેતુ	Consistent block time જાળવવું

Mining ના ફાયદા:

- **Financial Reward**: Successful mining માટે Bitcoin કમાવવું
- **Network Security**: વધુ miners = વધુ secure network
- **Transaction Processing**: Bitcoin transfers શક્ય બનાવવું
- **Decentralization**: Central authority ની જરૂર નથી

મેમરી ટ્રીક

Mining = પૈસા, Math, Maintenance

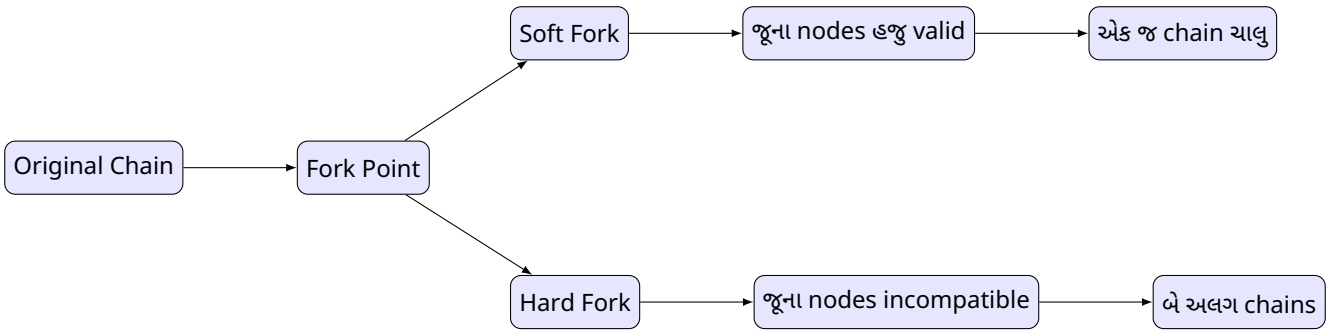
OR

પ્રશ્ન 3(અ) [3 ગુણ]

Soft fork અને Hard fork નો તફાવત આપો.

Fork સરખામણી:

બાબત	Soft Fork	Hard Fork
Compatibility	Backward compatible	Backward compatible નથી
Rules	Rules વધુ સખત બનાવે	Rules સંપૂર્ણ બદલે
Node Updates	જૂના nodes માટે વૈકલ્પિક	બધા nodes માટે ફરજિયાત
Chain Split	કાયમી split નથી	કાયમી split કરી શકે
Consensus	Implement કરવું સરળ	Majority agreement જરૂરી
ઉદાહરણો	SegWit (Bitcoin)	Bitcoin Cash, Ethereum Classic



આકૃતિ 13. Soft Fork vs Hard Fork Outcome

- **જોખમ**: Hard forks community split કરી શકે અને competing currencies બનાવી શકે
- **સુરક્ષા**: Soft forks સામાન્ય રીતે સુરક્ષિત અને ઓછા disruptive

મેમરી ટ્રીક

Soft = સમાન દિશા, Hard = મોટો તફાવત

OR

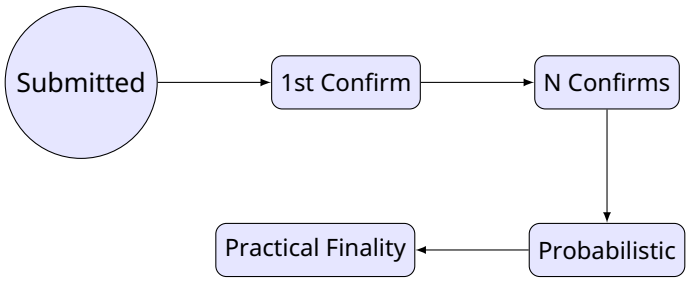
પ્રશ્ન 3(બ) [4 ગુણ]

બ્લોકચેનની દુનિયામાં Finality નાં શું મહત્વ છે?

Finality વ્યાખ્યા: એક વાર transaction confirm થઈ ગયા પછી તે reverse કે alter ન થઈ શકે તેની ગેરંટી.

મહત્વ:

બાબત	મહત્વ
Trust	Users ને વિશ્વાસ કે transactions કાયમી છે
Business Use	Companies completed transactions પર ભરોસો કરી શકે
Legal Certainty	Courts blockchain records enforce કરી શકે
Settlement	Financial institutions payments clear કરી શકે



આકૃતિ 14. Consensus and Finality Process

- **Bitcoin**: 6 confirmations સામાન્ય રીતે final ગણાય
- **Ethereum**: Proof of Stake સાથે ઝડપી finality તરફ જતું

મેમરી ટ્રીક

Final = હંમેશ માટે, મહત્વપૂર્ણ = પાછું ન બદલાય

OR

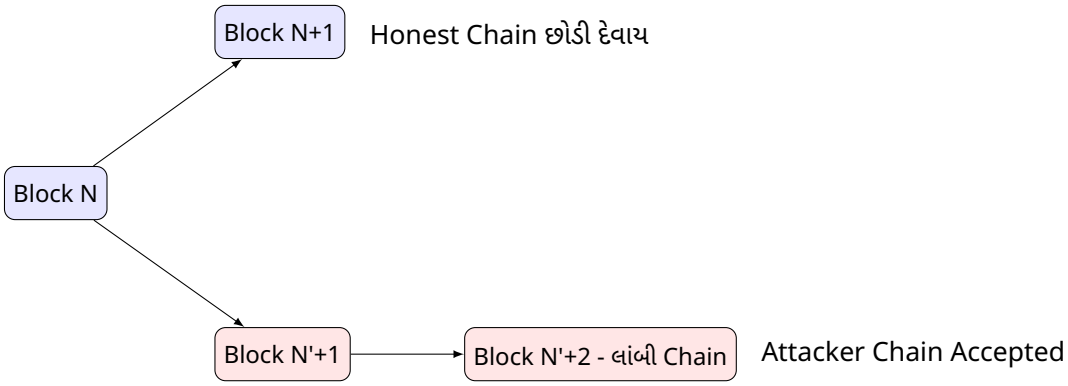
પ્રશ્ન 3(ક) [7 ગુણ]

બ્લોકચેનમાં 51% attack શું છે? ટૂંકમાં સમજાવો.

51% Attack વ્યાખ્યા: જ્યારે કોઈ એક entity network ની 50% થી વધુ mining power અથવા validators ને control કરે અને blockchain manipulate કરી શકે.

Attack પદ્ધતિ:

1. **Control**: >50% mining power મેળવવું
2. **Double Spend**: ગુપ્ત chain બનાવવી (alternative history)
3. **Execute**: લાંબી chain release કરવી
4. **Profit**: Coins બે વાર spend કરવા



આકૃતિ 15. 51% Attack: Longest Chain Rule Abuse

બચાવના પદ્ધતિઓ:

પદ્ધતિ	કેવી રીતે મદદ કરે
Decentralization	Mining ઘણા participants માં વહેંચવું
High Hash Rate	Attack ને economically અશક્ય બનાવવું
Proof of Stake	Attackers ના staked coins ગુમાવવા

મેમરી ટ્રીક

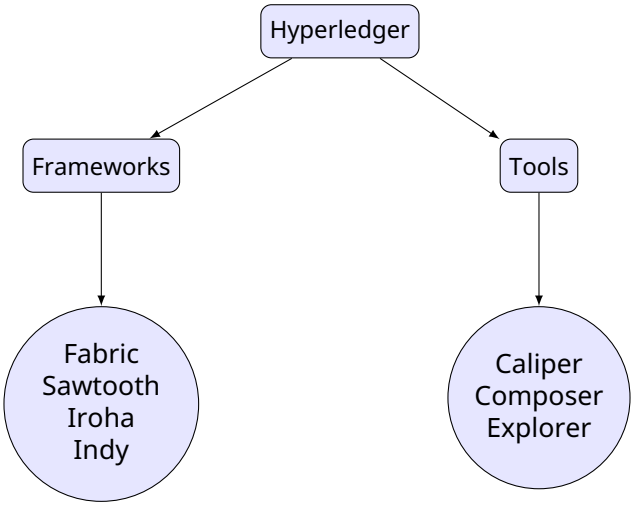
51% = બહુમતીની બદમાશી, Control = કોલાહલ

પ્રશ્ન 4(અ) [3 ગુણ]

વિવિધ પ્રકારના Hyperledger પ્રોજેક્ટ્સનાં વર્ણન કરો.

Hyperledger Project Types:

Project	હેતુ	Use Case
Fabric	Modular blockchain platform	Enterprise applications
Sawtooth	Scalable blockchain suite	Supply chain, IoT
Iroha	Mobile-focused blockchain	Identity management
Indy	Digital identity platform	Self-sovereign identity
Besu	Ethereum-compatible client	Public/private Ethereum
Burrow	Smart contract platform	Permissioned networks



આકૃતિ 16. Hyperledger Ecosystem

- **ફોકસ**: Enterprise અને business blockchain solutions
- **Open Source**: બધા projects મુફ્તમાં ઉપલબ્ધ

મેમરી ટ્રીક

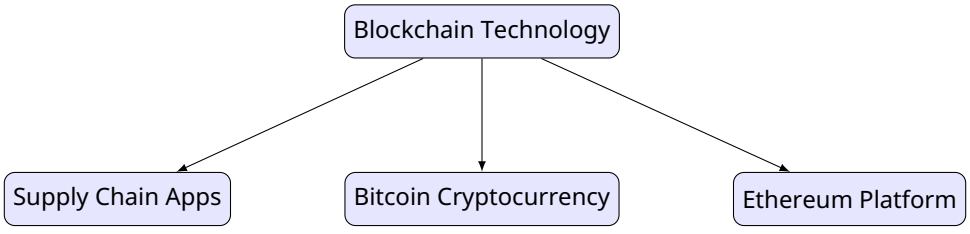
Hyperledger = High Performance, Low Publicity

પ્રશ્ન 4(બ) [4 ગુણ]

Blockchain અને Bitcoin નો તફાવત આપો.

વ્યાપક સરખામણી:

બાબત	Blockchain	Bitcoin
વ્યાખ્યા	Technology/Platform	Digital Currency
અવકાશ	વ્યાપક concept	Specific application
હેતુ	Record keeping system	Peer-to-peer payments
Applications	ઘણા industries	મુખ્યત્વે financial
લવચિકતા	Customize કરી શકાય	Fixed protocol



આકૃતિ 17. Blockchain vs Bitcoin Relationship

- **સમાનતા**: Blockchain ઈન્ટરનેટ જેવું, Bitcoin email જેવું
- **નિર્ભરતા**: Bitcoin ને blockchain જોઈએ, પણ blockchain ને Bitcoin જરૂરી નથી

મેમરી ટ્રીક

Blockchain = Building Block, Bitcoin = Specific Brick

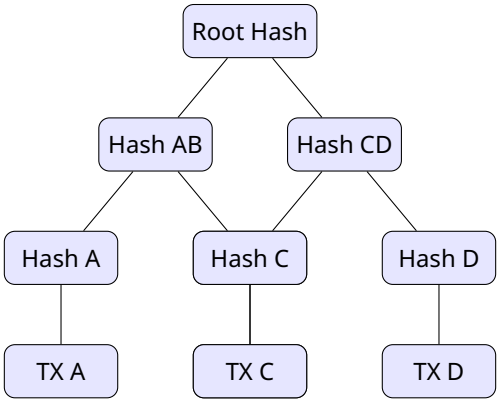
પ્રશ્ન 4(ક) [7 ગુણ]

ટૂંકી નોંધ લખો: Merkle Tree

Merkle Tree વ્યાખ્યા: Binary tree structure જેમાં દરેક leaf transaction hash દર્શાવે અને દરેક internal node તેના children નો hash ધરાવે.

Structure અને Components:

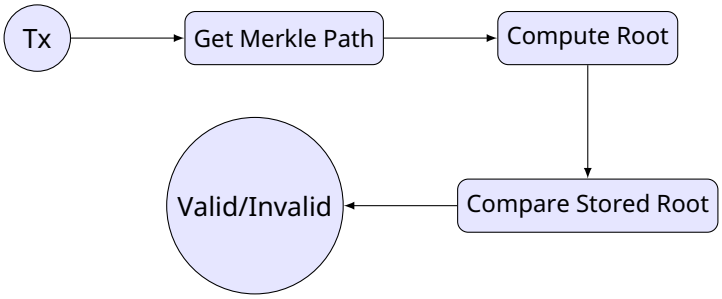
- **Leaf Nodes**: Individual transaction hashes
- **Internal Nodes**: બે child nodes નો hash
- **Root Hash**: આખા tree નો single hash



આકૃતિ 18. Merkle Tree Structure

ફાયદા:

- **Efficiency**: બધા data download કર્યા વગર ઝડપી verification
- **Security**: કોઈપણ change તુરંત detect થાય
- **Storage**: Block header માં ફક્ત root hash જરૂરી



આકૃતિ 19. Verification Process

મેમરી ટ્રીક

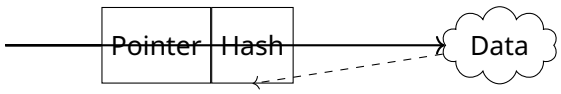
Merkle = Many Made One, Tree = Trustworthy

OR

પ્રશ્ન 4(અ) [3 ગુણ]

Hash pointer વિશે ટૂંકમાં ચર્ચા કરો અને Merkle tree માં તેનો ઉપયોગ કેવી રીતે થાય છે.

Hash Pointer વ્યાખ્યા: Data structure જેમાં data નું location અને તે data નો cryptographic hash બંને હોય.



આકૃતિ 20. Hash Pointer Concept

Merkle Tree માં ઉપયોગ:

- **Leaf Level**: Transaction ને point કરે, transaction hash ધરાવે
- **Internal Nodes**: Children ને point કરે, combined hash ધરાવે
- **Root**: Tree structure ને point કરે, overall hash ધરાવે

મેમરી ટ્રીક

Hash Pointer = સ્થાન + Verification

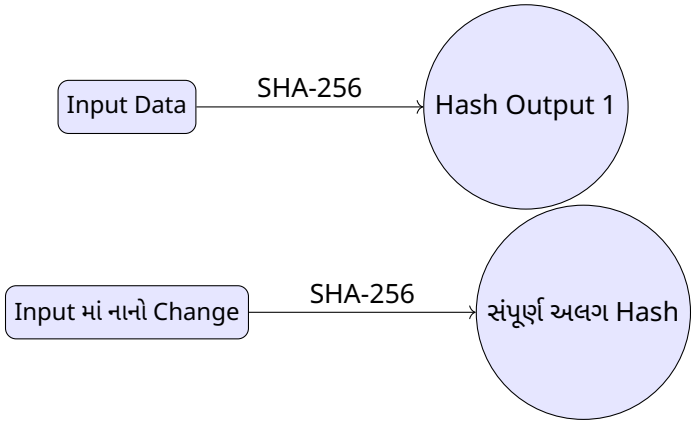
OR

પ્રશ્ન 4(બ) [4 ગુણ]

બ્લોકચેનમાં Hashing શું છે? Bitcoin માં તે કેવી રીતે ઉપયોગી છે?

Hashing વ્યાખ્યા: Mathematical function જે input data ને fixed-size characters ના string માં convert કરે.

Property	વર્ણન
Deterministic	સમાન input હંમેશા સમાન output આપે
Fixed Size	Output હંમેશા સમાન length (SHA-256 માટે 256 bits)
Avalanche Effect	નાનો input change = સંપૂર્ણ અલગ output
One-way	Original input શોધવા માટે reverse કરી શકાતું નથી



આકૃતિ 21. Avalanche Effect in Hashing

- **Algorithm**: Bitcoin SHA-256 hashing વાપરે
- **સુરક્ષા**: Blockchain ને tamper-evident બનાવે

મેમરી ટ્રીક

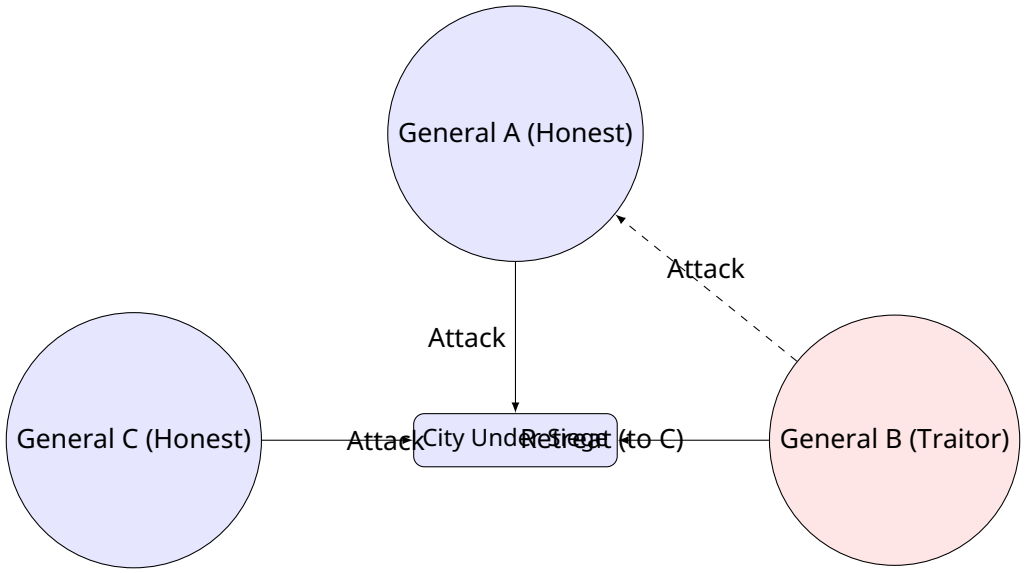
Hash = Fingerprint, Bitcoin = Hashing પર આધારિત

OR

પ્રશ્ન 4(ક) [7 ગુણ]

Classic Byzantine generals problem અને Practical Byzantine Fault Tolerance ને વિગતવાર સમજાવો.

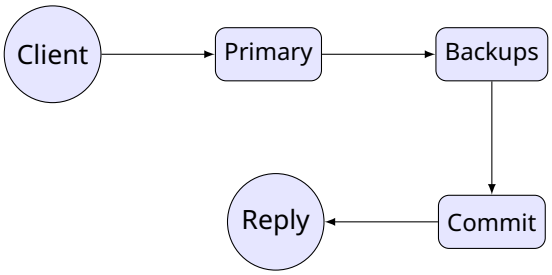
Byzantine Generals Problem: Distributed systems માં unreliable participants સાથે consensus achieve કરવાની સમસ્યા.



આકૃતિ 22. Byzantine Generals Problem

Practical Byzantine Fault Tolerance (pBFT):

- **Pre-prepare**: Leader proposal broadcast કરે
- **Prepare**: Nodes validate કરે અને agreement broadcast કરે
- **Commit**: Nodes decision પર commit કરે



આકૃતિ 23. pBFT Process Flow

મેમરી ટ્રીક

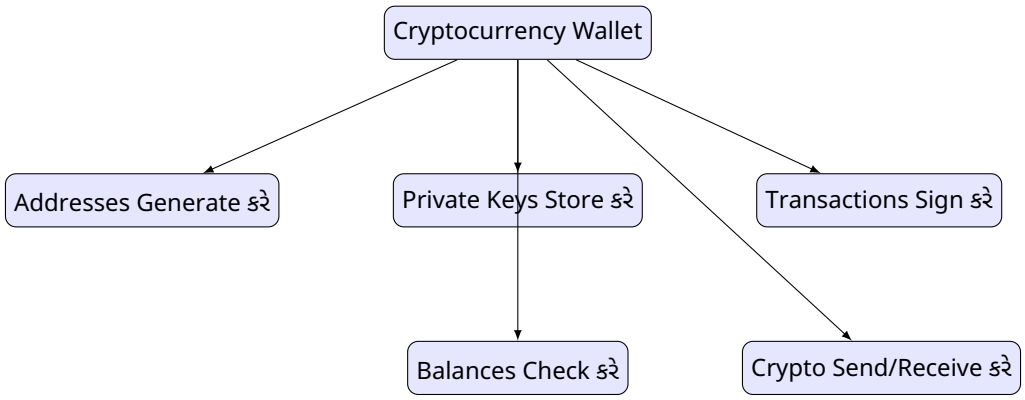
Byzantine = Bad actors, pBFT = Practical Fix

પ્રશ્ન 5(અ) [3 ગુણ]

બ્લોકચેનમાં cryptocurrency wallets ની યાદી બનાવો અને સમજાવો.

Cryptocurrency Wallet પ્રકારો:

Wallet Type	વર્ણન	Security Level
Hardware Wallet	Keys store કરતા physical device	ખૂબ ઊંચી
Software Wallet	Computer/phone પર application	મધ્યમ થી ઊંચી
Paper Wallet	કાગળ પર છપાયેલી keys	ઊંચી (સુરક્ષિત રીતે stored હોય તો)
Web Wallet	Online wallet service	મધ્યમ



આકૃતિ 24. Functions of a Wallet

મેમરી ટ્રીક

Wallet = Key Keeper, Not Coin Container

પ્રશ્ન 5(બ) [4 ગુણ]

ERC-20 ટોકનના ફાયદા અને ગેરફાયદા લખો.

ERC-20 Token વ્યાખ્યા: Ethereum blockchain પર tokens બનાવવા માટેનો standard protocol.

ફાયદા:

ફાયદો	લાભ
Standardization	બધા tokens સમાન રીતે કામ કરે
Interoperability	બધા Ethereum wallets સાથે compatible
Easy Development	નવા tokens બનાવવા સરળ
Wide Support	Exchanges અને services દ્વારા support

ગેરફાયદા:

ગેરફાયદા	સમસ્યા
Gas Fees	Network congestion દરમિયાન મોંઘા transactions
Scalability	Ethereum ની transaction throughput દ્વારા મર્યાદિત
Security Risks	Smart contract bugs થી token loss
Centralization	ઘણા tokens નું centralized control

મેમરી ટ્રીક

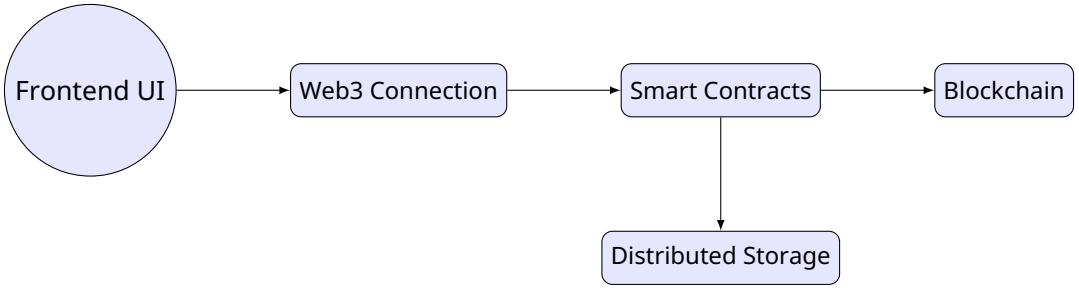
ERC-20 = Easy અને Expensive

પ્રશ્ન 5(ક) [7 ગુણ]

dApps નો ઉપયોગ શેના માટે થાય છે? dApps ના ફાયદા અને ગેરફાયદા સમજાવો.

dApps વ્યાખ્યા: Decentralized Applications જે blockchain networks પર central authority વગર run થાય.
dApps ઉપયોગ વર્ગીકરણ:

વર્ગ	ઉદાહરણો	હેતુ
DeFi	Uniswap, Compound	Financial services
Gaming	CryptoKitties	Blockchain games
Social Media	Steemit	Censorship-resistant platforms
Marketplaces	OpenSea	NFT trading



આકૃતિ 25. dApp Architecture

ફાયદા:

- **Censorship Resistance**: કોઈ એક control point નથી
- **Transparency**: Code અને data publicly verifiable
- **No Downtime**: ઘણા nodes માં distributed

ગેરફાયદા:

- **Poor User Experience**: જટિલ interfaces, ધીમા transactions
- **High Costs**: દરેક interaction માટે gas fees
- **Immutable Bugs**: Smart contract errors સહેલાઈથી fix ન કરી શકાય

મેમરી ટ્રીક

dApps = Decentralized but Difficult

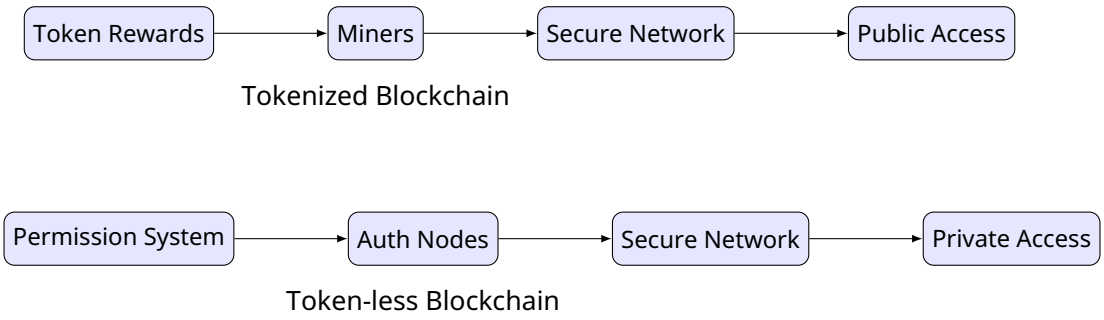
OR

પ્રશ્ન 5(અ) [3 ગુણ]

Tokenized અને token less બ્લોકચેનને વિગતવાર સમજાવો.

સરખામણી કોષ્ટક:

બાબત	Tokenized	Token-less
Incentive Model	Economic rewards	Permission-based
Access	Tokens હોય તો કોઈપણ	Restricted access
Governance	Token holder voting	Centralized control
Use Case	Public networks	Private/enterprise
Security	Economic game theory	Traditional security



આકૃતિ 26. Tokenized vs Token-less Architecture

મેમરી ટ્રીક

Token = Public Participation, Token-less = Private Permission

OR

પ્રશ્ન 5(બ) [4 ગુણ]

Hyperledger ના ફાયદા અને ગેરફાયદા લખો.

Hyperledger વ્યાખ્યા: Enterprise-grade blockchain solutions develop કરવા માટેનું open-source collaborative framework.

ફાયદા:

- **Enterprise Focus**: Business use cases માટે design
- **Modular Architecture**: જરૂર પ્રમાણે components customize કરી શકાય
- **Privacy**: Confidential transactions શક્ય
- **Permissioned Network**: કોણ participate કરી શકે તેનું control

ગેરફાયદા:

- **Centralization**: Public blockchains કરતાં ઓછું decentralized
- **Complexity**: Implement કરવા માટે technical expertise જરૂરી
- **No Token Economy**: Crypto incentives leverage કરી શકાતું નથી

મેમરી ટ્રીક

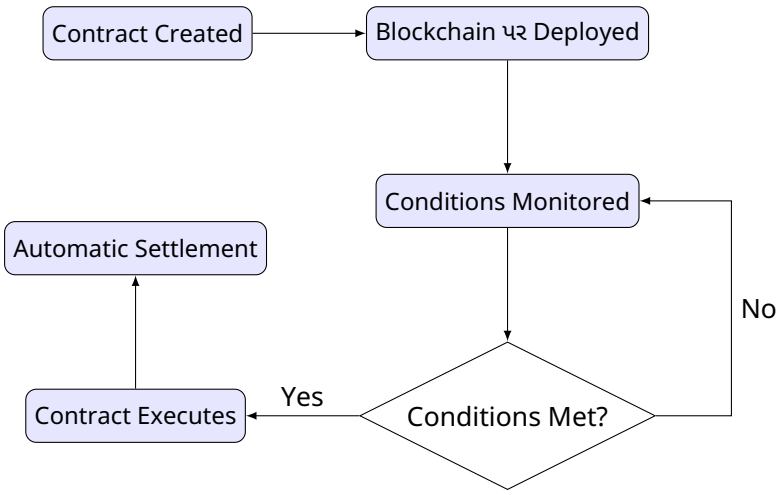
Hyperledger = High Performance, Low Publicity

OR

પ્રશ્ન 5(ક) [7 ગુણ]

Smart contract સમજાવો. Smart contract ની વિવિધ એપ્લિકેશન્સ લખો.

Smart Contract વ્યાખ્યા: Self-executing contracts જેના terms સીધા code માં લખાયેલા હોય અને blockchain પર આપોઆપ enforce થાય.



આકૃતિ 27. Smart Contract Workflow

Industry પ્રમાણે Applications:

Industry	Application	ફાયદો
Finance	Automated loans	ઝડપી, ઓછી costs
Real Estate	Property transfers	ફ્રોડ ઘટાડવું
Supply Chain	Product tracking	પારદર્શિતા
Healthcare	Insurance claims	Privacy protection

મેમરી ટ્રીક

Smart Contract = Self-executing, Solves Problems