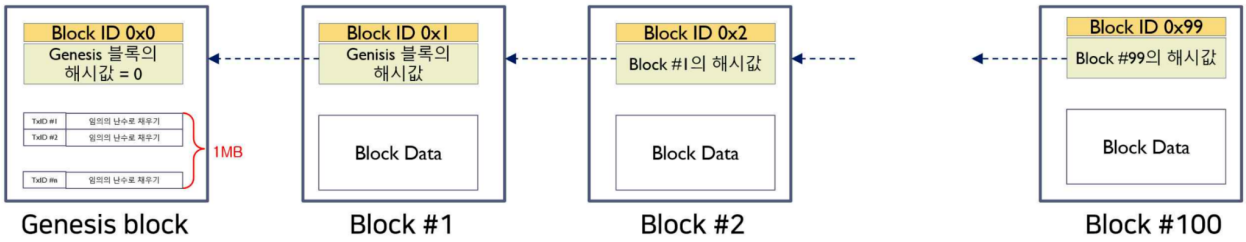
Questions

1. 아래 Diagram과 제시된 constraint를 만족하는 blockchain model를 구현하세요.
2. 위에서 구현한 blockchain을 개선하기 위하여, faster hash verification method and structure를 제시하고 이를 구현하세요.
3. 구현한 blockchain model를 사용하여, 특정 block의 transaction data가 위/변조 되었을 때 해당 block이 변경되었음을 확인하세요.

Diagram



<A simple blockchain structure>

Constraints

- Constraint 1: Block structure
  - 각 block은 Block ID와 이전 블록의 hash value, 그리고 block data로 구성됩니다.
  - Block 내에 있는 block data는 transaction들로 구성되며, 총 크기는 1MB입니다.
  - Block data를 구성하는 각 transaction은 160 bit size의 TxID 값과, (편의상 random하게 생성된) 864비트 크기를 갖는 transaction value로 구성됩니다.
  - 각 block에 포함되는 block data의 크기는 1MB가 되어야 합니다.
- Constraint 2: Block ID convention
  - Genesis block의 Block ID는 0x0입니다. 그 후 연결되는 block 의 Block ID 값은 0x1, 0x2, 0x3.. 처럼 1씩 increment합니다.
  - 이 때, Block ID의 크기는 160 bit입니다.
- Constraint 3: Block Hash generation
  - Genesis 블록의 initial hash value는 0으로 정하며, hash value의 크기는 160 bit입니다.
  - Block #1에 포함되는 hash value는 이전 block(genesis block) 전체에 대한 hash value입니다.
  - (n >= 2) Block #n는 Block #(n-1) 전체에 대한 hash value를 가집니다. 이 때, 각 block 내부의 hash value는 sha3-256의 256 bit output에서 하위 160 bit를 사용합니다.
- Constraint 4: Hash value generation
  - Hash value 생성시 사용하는 hash function은 반드시 sha3-256를 사용합니다.
  - Sha3-256 hash value에서 하위 160 bit를 사용합니다.
- Constraint 5: Transaction verification method
  - 특정 block의 transaction value가 위/변조 되었는지를 빠르게 검증할 수 있는 방법(faster hash verification method)를 고안하여 구현하세요.
- Constraint 6: Misc
  - 구현된 기능을 명확하게 파악할 수 있는 test code를 제시하여야 합니다.

- Consensus algorithm에 대한 고려는 전혀 할 필요가 없습니다.