

Claims

1. A computer-implemented method of pseudo-randomly generating winning game elements for use in playing a game, wherein the game comprises a set of first game elements used to determine an outcome of the game, wherein the game is played by one or more respective users, and wherein the method is performed by an oracle and comprises:
 - obtaining a set of seed data items, wherein the set of seed data items comprises one or more user seed data items generated by a respective user;
 - obtaining a sequence of first public keys, each first public key representing a respective one of the set of first game elements;
 - generating a first output of a game transaction, wherein the game transaction is a blockchain transaction, wherein the first output comprises an output script comprising the sequence of at least some of the first public keys, and wherein the output script is configured to, when executed, generate at least one first pseudorandom number, the at least one first pseudorandom number being based on the set of seed data items, and to select at least one first winning key, the at least one first winning public key being the public key at a position in the sequence of first public keys corresponding to the at least one first pseudorandom number.
2. The method of claim 1, comprising transmitting the game transaction to one or more of the respective users and/or the blockchain network.
3. The method of claim 1 or claim 2, wherein the set of seed data items comprises an oracle seed data item generated by the oracle.
4. The method of any preceding claim, comprising:
 - generating a commitment transaction, wherein the commitment transaction is a blockchain transaction comprising the oracle seed data item; and
 - transmitting the commitment transaction to a current one of the respective users.
5. The method of any of claims 1 to 3, comprising:

obtaining a commitment transaction, wherein the commitment transaction comprises the user seed data item generated by the current user; and
transmitting the commitment transaction to a blockchain network for inclusion in a blockchain.

6. The method of claim 4 or claim 5, wherein the game transaction comprises a first input configured to unlock a first output of the commitment transaction.

7. The method of any of claims 1 to 6, wherein the first pseudorandom number is generated by:

applying a first hash function to a combination of the set of seed data items to generate a first hash result; and
mapping the first hash result to a number based on a first total number of first game elements in the set of first game elements.

8. The method of claim 7, wherein said mapping comprises taking a first modulus of the first hash result, wherein said first total number is the first modulus.

9. The method of any preceding claim, comprising:

generating a mapping hash by applying a hash function to a mapping on the sequence of public keys and the respective first game elements represented by those public keys; and
making available the mapping hash to at least the current user.

10. The method of claim 9, wherein said making available of the mapping hash comprises including the mapping hash in a transaction for inclusion in the blockchain.

11. The method of any preceding claim, wherein said obtaining of the set of seed data items comprises obtaining the respective user seed data items from respective users.

12. The method of any preceding claim, wherein said obtaining of the sequence of first public keys comprises obtaining one or more of said first public keys from the current user.

13. The method of any preceding claim, wherein the first output script is configured to, when executed, lock the first output to the first winning public key.
14. The method of claim 13, comprising, transmitting, to the current user, a private key corresponding to the first winning public key.
15. The method of any of claims 1 to 12, wherein the first output script is configured to, when executed, lock the first output to a) one or more user public keys associated with the current user, or b) an oracle public key associated with the oracle, wherein the first output is locked to either a) the one or more user public keys, or b) the oracle public key, based on the first winning public key.
16. The method of any of claims 1 to 12, wherein the game comprises a set of second game elements used to determine the outcome of the game, and wherein the method comprises:
- obtaining a sequence of second public keys, each second public key representing a respective one of the set of second game elements; and
 - wherein the output script is configured to, when executed, generate a second pseudorandom number, the second pseudorandom number being based on the set of seed data items, and to select, as a second winning public key, a public key at a position in the sequence of second public keys corresponding to the second pseudorandom number.
17. The method of claim 16, wherein the second pseudorandom number is generated by:
- applying a second hash function to a combination of the set of seed data items to generate a second hash result; and
 - mapping the second hash result to a number based on a second total number of second game elements in the set of second game elements.

18. The method of claim 17, wherein said mapping of the second hash result comprises taking a second modulus of the second hash result, wherein said second total number is the second modulus.

19. The method of any of claims 16 to 18, wherein the first output script is configured to, when executed, lock the first output to a) one or more user public keys associated with the current user, or b) an oracle public key, wherein the first output is locked to either a) the one or more user public keys, or b) the oracle public key, based on the first winning public key corresponding to at least the second winning public key.

20. The method of claim 19, wherein the first output script comprises a multi-signature script for locking the first output to a) one or more user public keys.

21. The method of any preceding claim, wherein one or more first public keys represent the same first game element, and/or wherein one or more second public keys represent the same second game element.

22. A blockchain transaction for inclusion in a blockchain, the transaction comprising:
an output, wherein the output comprises an output script comprising a sequence of public keys, each public key representing a respective one of a set of game elements, and wherein the output script is configured to generate at least one first pseudorandom number, the at least one first pseudorandom number being based on a set of seed data items, wherein the set of seed data items comprises one or more user seed data items generated by a respective user, and wherein the output script is configured to select at least one first winning key, the at least one first winning public key being the public key at a position in the sequence of public keys corresponding to the at least one first pseudorandom number.

23. A computer-readable storage medium having stored thereon the transaction of claim 22.

24. Computer equipment comprising:

memory comprising one or more memory units; and

processing apparatus comprising one or more processing units, wherein the memory stores code arranged to run on the processing apparatus, the code being configured so as when on the processing apparatus to perform the method of any of claims 1 to 21.

25. A computer program embodied on computer-readable storage and configured so as, when run on computer equipment, to perform the method of any of claims 1 to 21.