

1 緒言

各塾の調査では中学、高校で学習する科目の中で数学と英語は苦手になりやすいと言われている [1]. この 2 科目の共通点である積み上げ型教科という点に注目した. 科目には独立型教科と積み上げ型教科の 2 種類に分けられると言われている. 独立型教科とは単元ごとに独立していて, 単元ごとの関連性があまりない教科を指し, どの単元から学習しても支障が出ない. それに対して積み上げ型教科とは既に学習した知識を使うことを前提として次の単元を学習する教科で, ある程度決められた順番で学習する必要がある. 本研究では前提知識となる単元を「前提単元」, 前提知識を用いて学習を行う単元を「主単元」と称した. 前提単元には学習目的が主単元で利用することになっているものがあり, このような単元は学習目的や実用例などのイメージがしづらく, 理解の妨げになっていると考えた. また前提単元が理解できないと主単元にまで影響を及ぼしてしまい, 理解できない単元が増えるため, 苦手な教科になってしまう.

そこで前提単元の具体例がわかりづらく, 主単元の具体例がわかりやすいとき, 主単元の概要をあらかじめ学習することで, 理解を促進することができるという仮説を立てた.

本研究では, 積み上げ型教科の理解力向上を目的として, 講義にて学生を対象に実験を行い, この仮説を検証する.

2 教科の特性

学校で学習する教科には独立型教科と積み上げ型教科に分かれ，積み木のような図で表される。

2.1 独立型教科

独立型教科とは単元ごとに独立していて，単元ごとの関連性があまりない教科を指し，主に国語や社会が該当する。図4は社会の単元である「地理」「歴史」「公民」がそれぞれ独立していることを示している。

独立している各単元の中で積み上げ型教科の特性を示しているものや，独立している各単元で繋がりがああることもあるが，基本的にはどの単元から学習しても支障がない。

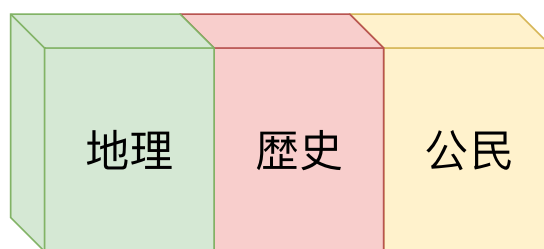


図1 独立型教科の例

2.2 積み上げ型教科

積み上げ型教科とは既に学習した知識を使うことを前提として次の単元を学習する教科で、主に数学や英語が該当する。図2では数学を例としてあげている。この例ではまず「正の数・負の数」を学習し、それを前提単元として「文字式」を学習する。次に学習した「文字式」を前提単元として「方程式」や「平面図形」を学習する。「方程式」「平面図形」をそれぞれ前提単元として「比例・反比例」「空間図形」を学習していく。積み上げ型教科ではこのように学習した内容を次の学習の基礎知識として用いるため、順番に学習していくことになる。

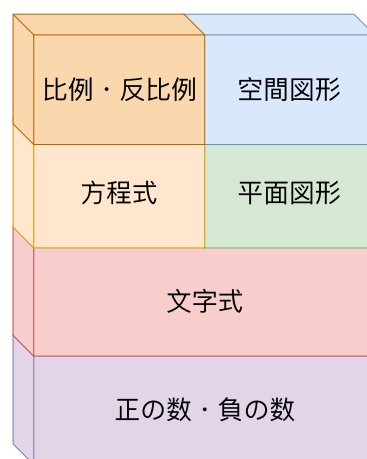


図2 積み上げ型教科の例

積み上げ型教科には大きく分けて3種類の形が存在する。図3では3種類の形を示す。

左側の図は1つの前提単元を用いて、1つの主単元を学習する形を表した図である。中央の図は1つの前提単元を用いて、複数の主単元を学習する形を表した図である。右側の図は複数の前提単元を用いて1つの主単元を学習する形を表した図である。

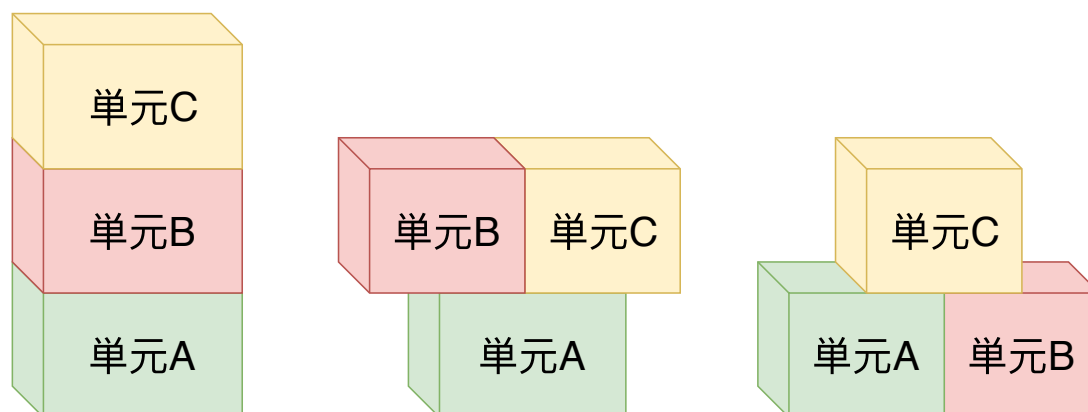


図3 積み上げ型教科の教育モデル

3 ハッシュ関数

任意のデータを固定長の値であるハッシュ値に置き換える関数である。ハッシュ関数は不可逆的な一方向関数であり、ハッシュ値から元のデータに復元することは不可能である。

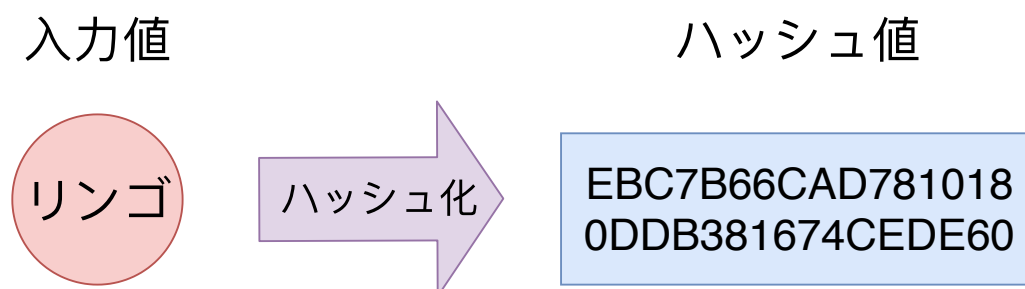


図4 ハッシュ化の例

3.1 ハッシュ関数の安全性

ハッシュ関数の安全性について以下の3種類の特徴がある。

衝突困難性（強衝突耐性）

同じハッシュ値を与える2つの入力値を求めることが困難であるような性質である。

第二原像計算困難性（弱衝突耐性）

与えられた入力値に対して、その入力値のハッシュ値と同じハッシュ値の入力値を求めることが困難であるような性質である。

原像計算困難性

与えられたハッシュ値に対して、そのハッシュ値を出力する入力値を求めることが困難であるような性質である。

一つのアルゴリズムについて違う入力値の同じハッシュ値が見つかることを強衝突耐性の突破と呼び、与えられた入力値と同じハッシュ値の入力値を求められるようになることを弱衝突耐性の突破と呼ぶ。

原像計算困難性を満たさないハッシュ関数では、任意の入力値からハッシュ値を得られるため、第二原像困難性を満たさない。また第二原像計算困難性を満たさないハッシュ関数では、衝突困難性を満たさない。したがって、

原像計算困難性 \supset 第2原像計算困難 \supset 衝突困難である。

3.2 ハッシュ関数のアルゴリズム

ハッシュ関数はハッシュ値のビット数や、仕組みによって様々な種類がある。以下はハッシュ関数の中でも代表的なアルゴリズムである。

MD5

1991 年に開発されたハッシュ値が 128 ビットのハッシュ関数の一つである。それほど性能の低いコンピュータでも数十分程度で同一ハッシュ値のデータ列が生成できるため、強衝突耐性が容易に突破される状態にある。しかし弱衝突耐性は容易に突破されることはない。

SHA-1

ハッシュ値が 160 ビットの Secure Hash Algorithm(SHA) シリーズのハッシュ関数の一つである。アメリカ合州国において機密情報を扱う際に法律によって要求されるハッシュアルゴリズムとして用いられていたが、2011 年に理論上ではあるが強衝突耐性を突破されているため、改良版である SHA-2 へ移行している。

SHA-2

SHA-1 が改良されて作られたハッシュ関数で、SHA-224, SHA-256, SHA-384, SHA-512 と呼ばれるハッシュ関数の総称である。またビット数はそれぞれ 224 ビット, 256 ビット, 384 ビット, 512 ビットであり、SHA-1 より長くなっている。その中でも SHA-256 は計算速度や安全性のバランスに優れていて、もっとも普及している。ビットコインに使われているのも SHA-256 である。

3.3 ハッシュ関数の実用例

3.3.1 ファイルの同一性確認

ハッシュ関数は改ざんや破損の検知など、ファイルの同一性確認に使用される。
図5は2つのハッシュ値を比較して一致した時と不一致の時の図である。

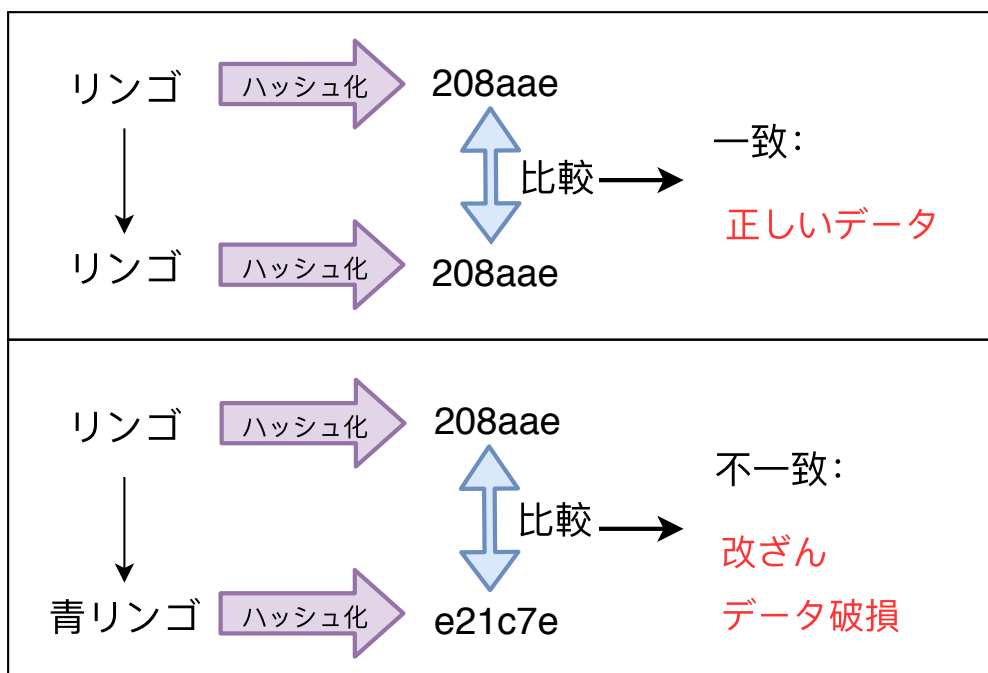


図5 ハッシュを用いたファイルの同一性確認の流れ

3.3.2 パスワードの管理

ハッシュ関数はWEBサイトなどでパスワードを用いる際に使われる。パスワードを平文で管理するとパスワードが漏洩したときに第三者に不正にアクセスされる危険がある。そのためパスワードをハッシュ化し、ハッシュ値の状態で管理することで漏洩しても不正ログインを防ぐことができる。

しかし同じデータをハッシュ化すると同じハッシュ値になることから、あらかじめ使われやすい単語などをハッシュ化しデータベース化し、不正ログインを試みようとするサイトがある。そのため以下のような手段が取られている。

salt を加える

平文のパスワードに salt と呼ばれるランダムな文字列を加えることによって、1つの文字列に対するハッシュ値の種類を増やすことで、データベースを作成しておく攻撃を困難にする。

ハッシュ関数を繰り返す

平文のパスワードをハッシュ化して出たハッシュ値を、さらにハッシュ化することによって新たなハッシュ値を生成することができる。

4 暗号

暗号とは通信を行う際、第三者にその内容を知られないようにするための手段である。元の文に一定の規則を用いて特定の変形を加えることを暗号化、暗号化する前の元の文を平文、暗号から平文に戻すことを復号と呼ぶ。暗号化する手段が暗号アルゴリズムであり、また暗号化や復号化に使うデータを鍵と呼ぶ。暗号は特性によって古典暗号と現代暗号に分かれる。アルゴリズムがわかれば解読が容易になる暗号を古典暗号、アルゴリズムは公開するが鍵を非公開にすれば安全な暗号を現代暗号という。

4.1 共通鍵暗号

暗号化と復号に同じ鍵を用いる暗号の方式である。通信相手に鍵を送り、その鍵を用いて情報を暗号化する。暗号化された情報が送られてきたら、同じ鍵を使って復号する。

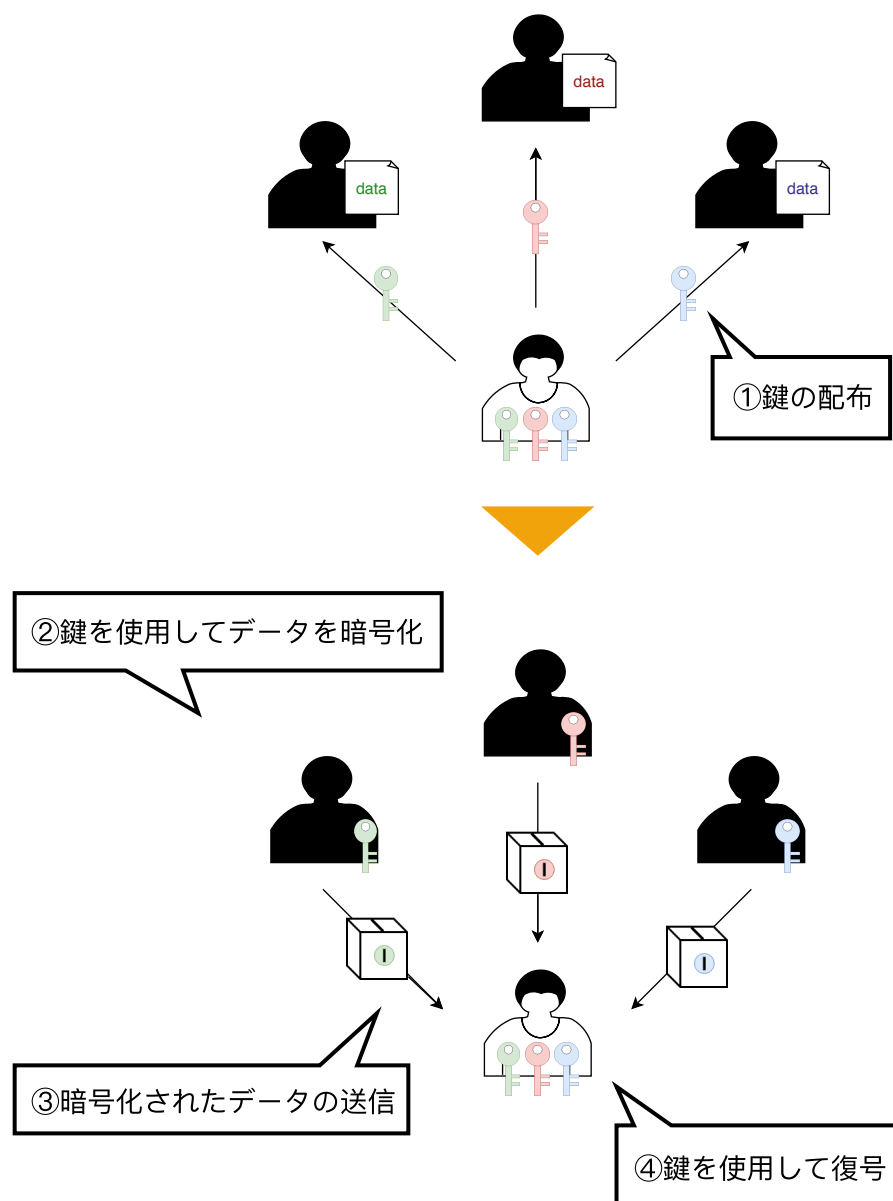


図6 共通鍵暗号通信の流れ

共通鍵暗号通信では暗号化と同じ鍵を使っているため、鍵を送る際に第三者に傍受される危険がある。また同じ鍵を複数のユーザーで使用すると、他のユーザーが復号する危険性があるため、ユーザーごとに鍵を生成する必要がある。

4.1.1 換字式暗号の仕組みと例

平文の文字を他の記号や文字に置き換えて暗号文を作る古典暗号の方式である。換字式暗号の代表としてシーザー暗号がある。シーザー暗号は鍵である決められた文字数分のアルファベットをずらして暗号化を行う。図7は鍵が1であるときの例である。

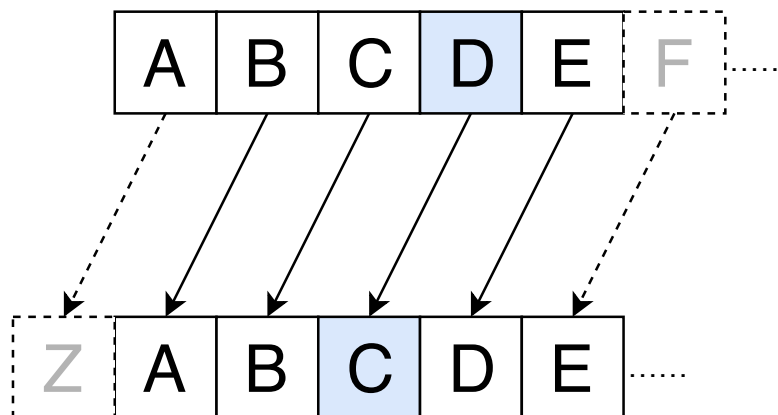


図7 シーザー暗号の例

4.1.2 転置式暗号の仕組みと例

平文の位置を並び替えて暗号文を作る古典暗号の方式である。転置式暗号の代表としてスキュタレー暗号がある。スキュタレー暗号はスキュタレーと呼ばれるシリンダーに細長い紙を巻きつけ、平文を横書きに書く。紙をスキュタレーから外すと順番が入れ替わった暗号ができる。暗号の受け取る側は同じ半径のスキュタレーを用意し、紙を巻きつけることで復号することができる。この場合、鍵は同じ直径であることである。

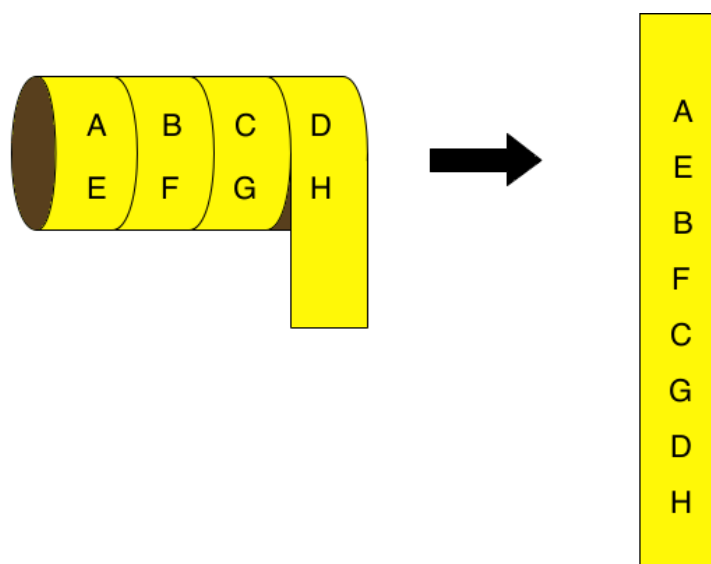


図8 スキュタレー暗号の例

4.2 公開鍵暗号

暗号化と復号に別の手順を用いる暗号方式である。ユーザは秘密鍵から公開鍵を生成する。通信相手に送る鍵を公開鍵と呼び、自分の手元に保管しておく鍵を秘密鍵と呼ぶ。公開鍵を通信相手に送り、通信相手は公開鍵を使って情報を暗号化する。ユーザは受け取った暗号化された情報を秘密鍵を用いて復号する。

公開鍵暗号では「閉じることにはできても開けることができないこと」を安全の根拠としており、一方向関数である素因数分解や楕円曲線が使われる。

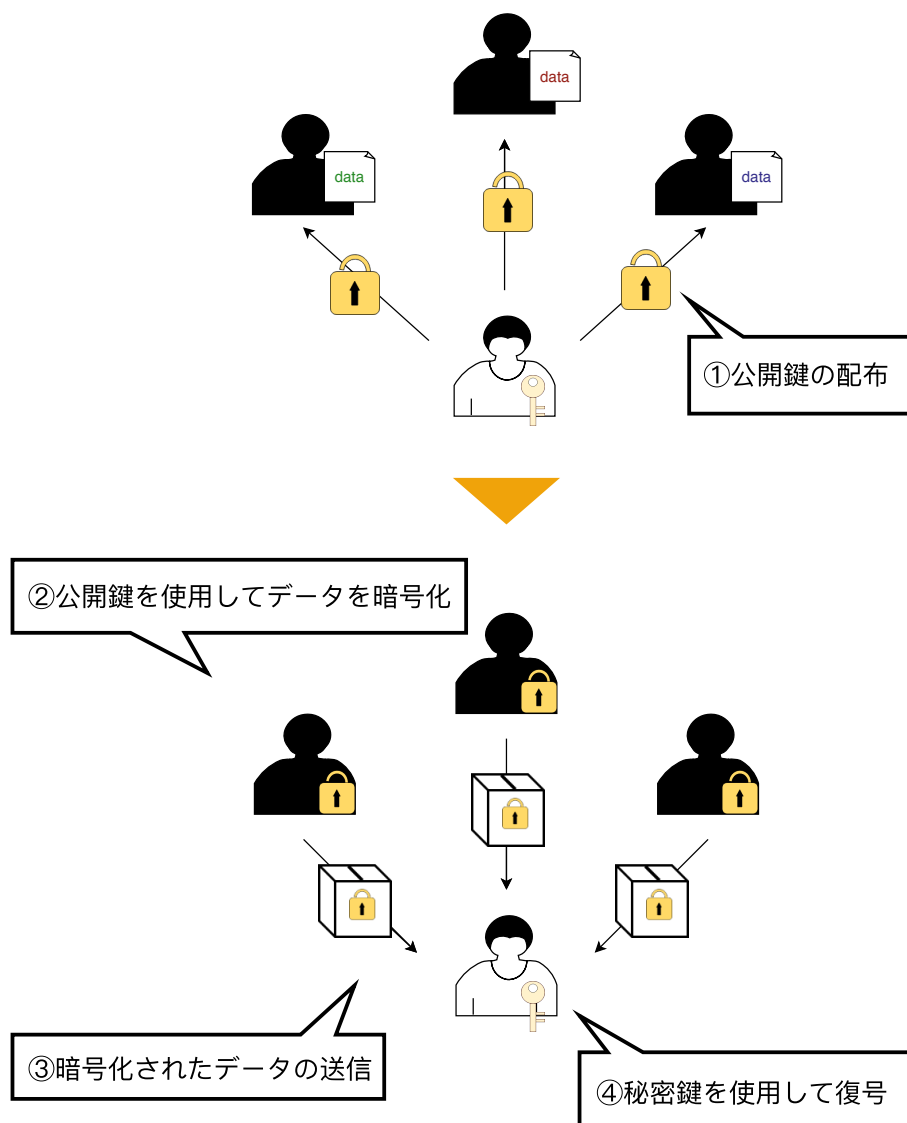


図9 公開鍵暗号通信の流れ

公開鍵暗号は共通鍵暗号とは違い各ユーザーごとに鍵を生成する必要がある。また鍵の受け渡しの際に傍受される可能性がある共通鍵暗号通信とは違い、公開鍵では復号することができない。そのため共通鍵暗号通信よりも安全性が高い。しかし復号に複雑な計算を用いるため負荷が大きくなるため通信に時間がかかるという欠点もある。

4.2.1 RSA 暗号

RSA 暗号とは現在インターネットで広く使われている公開鍵暗号の一つである。発明者である Ronald Linn Rivest, Adi Shamir, Leonard Max Adleman の頭文字をとって名付けられた。RSA 暗号は桁数の多い素数の掛け算をするのは簡単だが、その合成数の素因数分解をするのは困難であることを安全性の根拠としている。

4.2.2 電子署名の仕組み

RSA 暗号や楕円曲線 DSA など電子署名の役割を持つ暗号方式もある。これらの公開鍵と秘密鍵が同じ構造をしている暗号では、どちらの鍵を使っても暗号化することができる。そのため秘密鍵で暗号化し、公開鍵で復号することによって、送信者が本人であることを確認することができる。

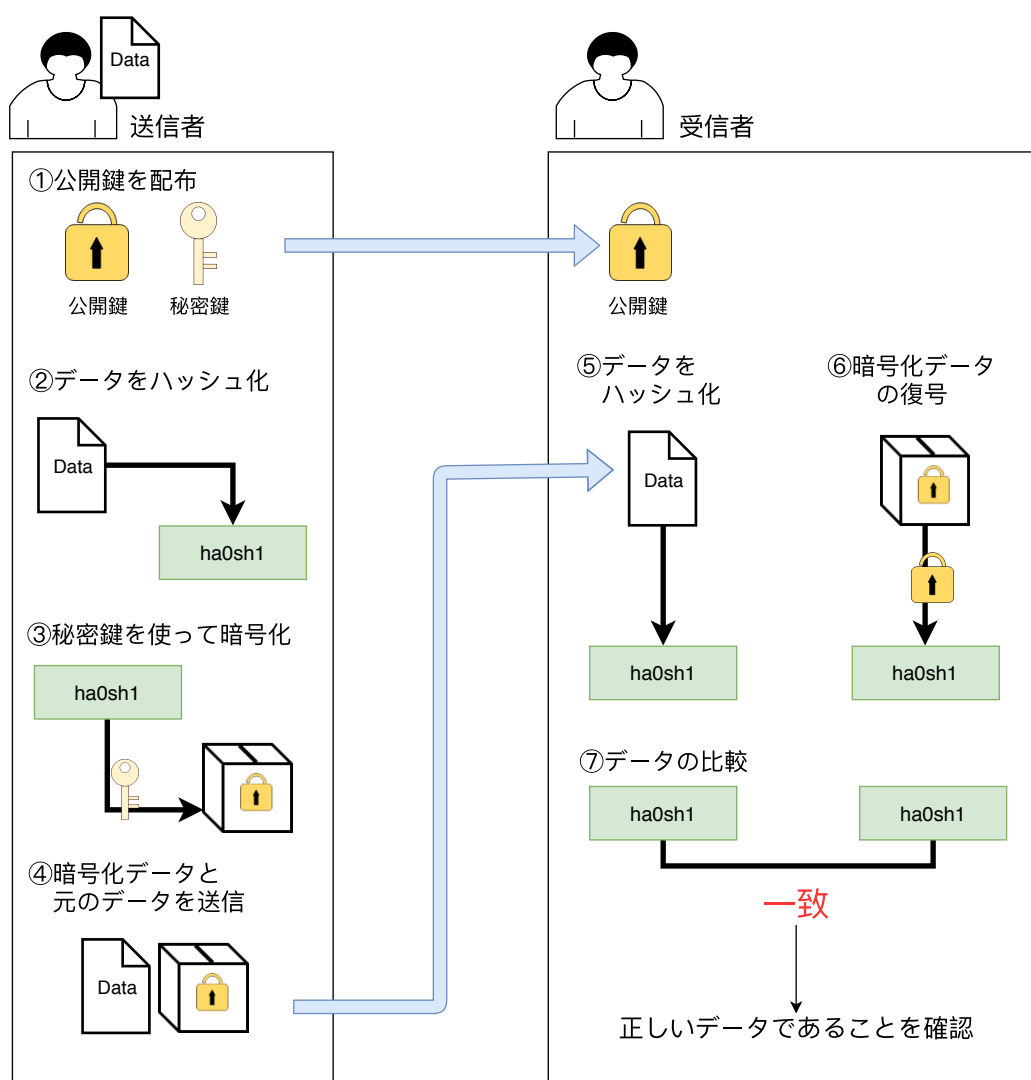


図 10 電子署名の流れ

4.3 SSL 暗号化通信

共通鍵暗号方式は公開鍵暗号方式よりも負荷が小さいが、鍵の受け渡し時に傍受される危険性があった。そこで鍵の受け渡し時に公開鍵暗号方式を用いることで安全性を確保したものが SSL 暗号化通信である。

4.3.1 SSL 暗号化通信の仕組み

SSL 暗号化通信では最初にユーザがサーバに接続要求を行い、送られてくる SSL サーバ証明書によって安全性を確認する。安全性を確認したら、共通鍵を公開鍵暗号通信を使ってサーバに送ることで安全に共通鍵の受け渡しを行うことができる。

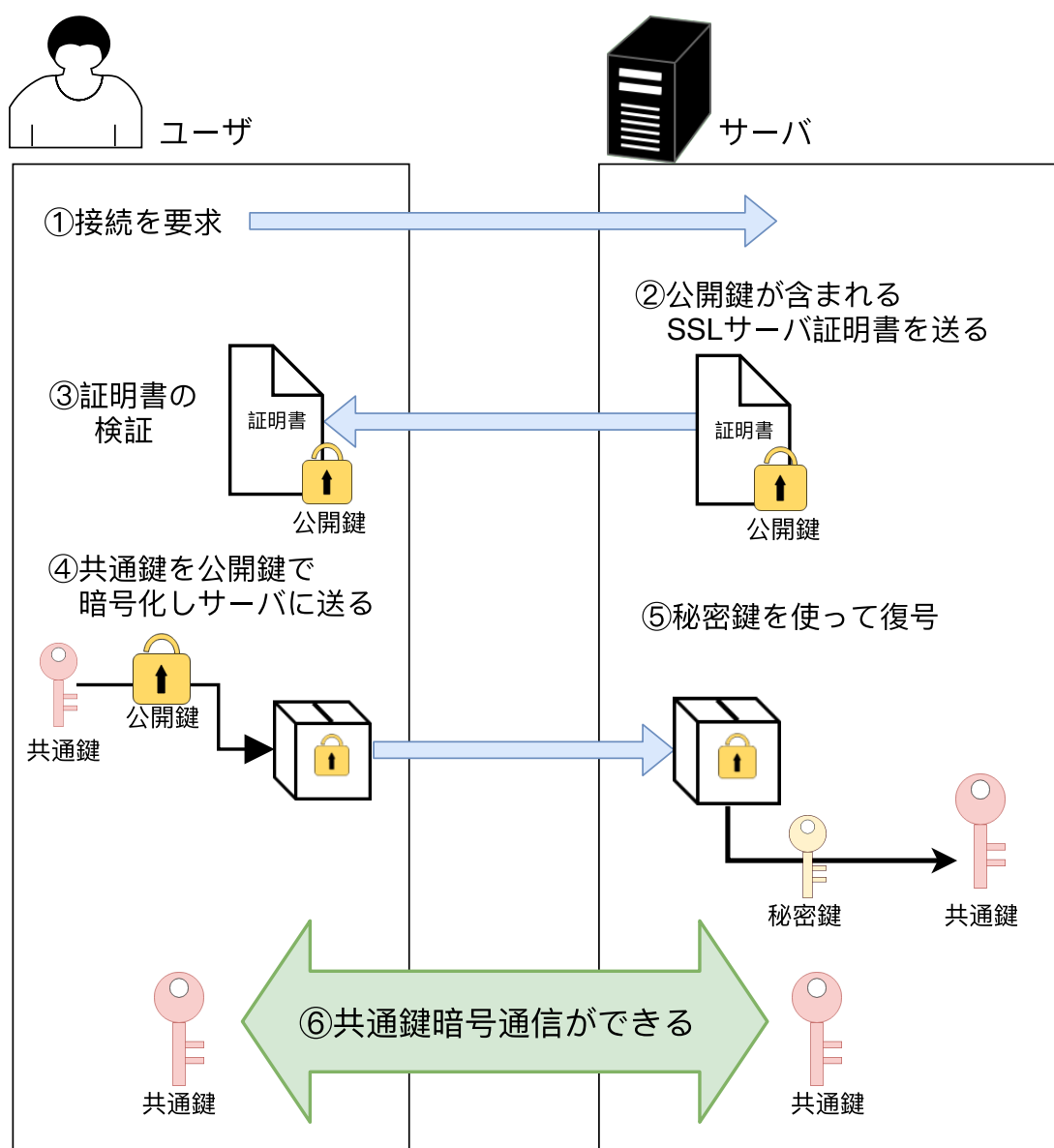


図 11 SSL 暗号化通信の流れ

5 ブロックチェーン

ブロックチェーンとは、2009 年に Satoshi Nakamoto 氏の論文 [2] で提唱された仮想通貨であるビットコインの根幹技術として生まれた分散型台帳システムである。

5.1 ブロックチェーンの特徴

取引を記録する台帳の 1 ページをブロックという。ブロックチェーンはインターネット上の複数のコンピュータでお互いに取引の記録を共有し、検証しながら正しい記録の入ったブロックを鎖のように繋げていく。

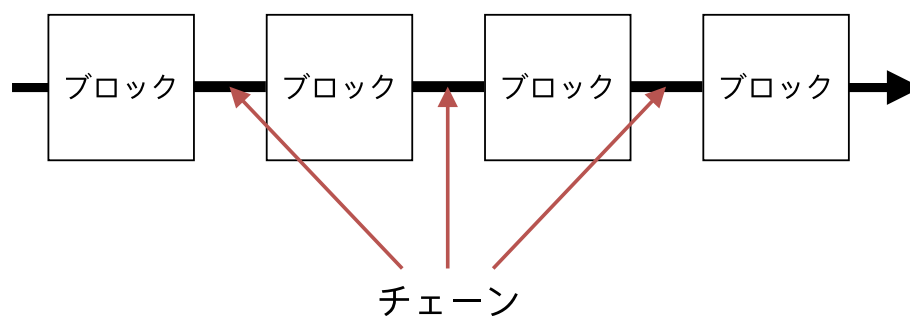


図 12 ブロックチェーンとは

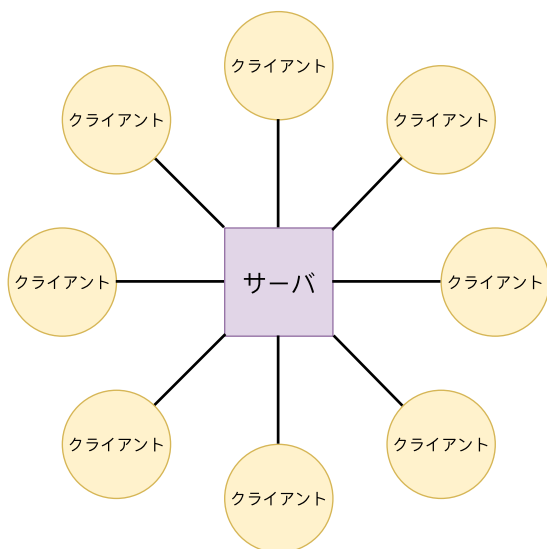
5.1.1 改ざんへの耐性がある

ブロックチェーンの各ブロックはハッシュ関数によって繋がれている。ハッシュは不可逆性を持っており、ハッシュ値から元のデータを復元することはできない。ブロックチェーンでは複数の取引データがハッシュ化によって 1 ブロックにまとめられ、次のブロックに追加される。そのため過去のブロックを改ざんすると次のブロックとの整合性が取れなくなるため改ざんが困難である。

5.1.2 P2P 分散型システム

ブロックチェーンは P2P（Peer to Peer）を用いて取引データをやりとりしている分散型のシステムである。P2P は特定の管理者がサーバを管理するクライアント・サーバ型とは違い、各ノード間で対等に直接通信を行い、ネットワークを形成するネットワークである。

クライアント・サーバ型



P2Pネットワーク型

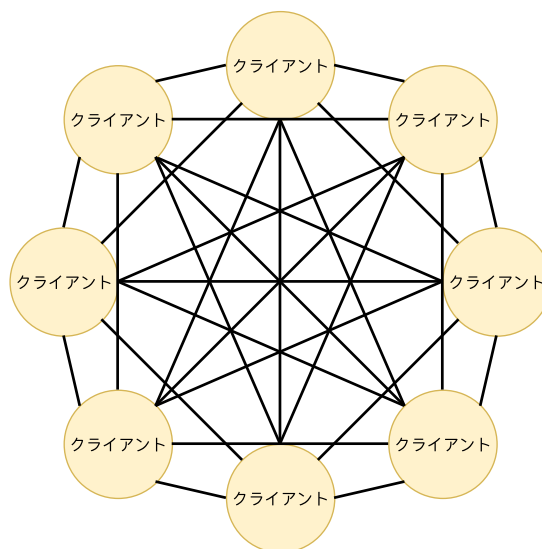


図 13 クライアント・サーバ型と P2P ネットワーク型の違い

ブロックチェーンは P2P ネットワーク型であるため、データをまとめて記録したり、管理をしている機関のない非中央集権的なシステムとなっている。そのため一部のコンピュータに障害が発生してもシステムを維持することができる。

5.1.3 ユーザとブロックチェーン間での通信

ユーザとブロックチェーン間での通信には公開鍵暗号とハッシュを用いた電子署名の技術が使われる。取引記録をハッシュ化したものを取引記録と一緒にブロックチェーンに送ることで、取引をしたユーザが本人であることを確認している。

5.1.4 コンセンサスアルゴリズム

チェーンにブロックを追加するにあたって、参加しているノードによる承認手続が必要となる。ブロックチェーンでは管理者がいがないため「不正」か「正当」の振り分けを、参加しているノードがある一定の条件のもとで行っている。またその一定の条件を満たしたノードが新たなブロックを作成する。

5.2 仮想通貨とは

一般的にはネットワーク上で電子的な決済の手段として流通しているが、法定通貨のように特定の国家による裏付けのないものである。ただし様々な種類の仮想通貨があり、定義や分類などは一様ではない。

5.3 コンセンサスアルゴリズムによるブロックチェーンの分類

5.3.1 Proof of Work (PoW)

PoW ではネットワークに参加しているノードは一番最初に nonce と呼ばれる値に当てはまる解を探す。この nonce を探す作業を採掘（マイニング）と呼ぶ。採掘に成功したノードはブロックを作成し、ブロックチェーンをつなげる権利を得ることができる。

ビットコインではこの PoW 方式が採用されている。ビットコインの採掘では報酬として計算量に応じたビットコインをもらうことができる。

PoW の問題点としては総計算能力の過半数を占めるノードを用いることで虚偽のブロックを有効にすることができる 51 %問題がある。また PoW で報酬を得るためには高性能のコンピュータを常に起動して計算作業をさせる必要があり、無駄に電気コストがかかるという欠点もある。

5.3.2 Proof of Stake (PoS)

PoS でもブロック追加のためにはハッシュ関数等の計算による解が必要となるが、PoW では計算量に応じた報酬を与えていたのに対し、仮想通貨の保有量に応じた報酬が与えられる。これは PoW の採掘に対し、鑄造（minting, forge）と呼ばれる。

また PoS では PoW と違い、計算量で争うことがなくなるため、PoW に比べ必要な電気コストが大幅に減らすことができる [3]。

PoW で 51 %攻撃をするためには膨大な計算能力を保有する必要があったが、PoS では仮想通貨の総発行枚数の 51 %以上を保有する必要があり、PoW よりもコストがかかる。また自分が多く保有している通貨に攻撃を仕掛けるため、攻撃者にとってはメリットがあまりない。

しかし PoS では通貨を保有することで報酬がもらえるという仕組みのため、通貨に必要な流動性が低くなってしまいう問題や流動性が低くなることで、仮想通貨の価格が低いときに大量に保有した人に報酬が集まり、貧富の差が広がるという問題がある。

5.3.3 Proof of Importance (PoI)

PoI では参加者の重要度によって発言権が付与される。重要度は仮想通貨の保有量と取引（仮想通貨の流動性を高めたか）によって決められる。PoS では富裕層が有利になるというデメリットがあったが、PoI 方式では流動性が高まるため、貧富の差が極端に広がることがない。PoI の仕組みによって報酬を得ることをハーベスティング（収穫）と呼ぶ。

しかしハーベスティングに参加するためには一定量の仮想通貨を保有する必要があり、PoS ほどではないが富裕層が力を持つのではないかという懸念がある。

5.4 Proof of Work におけるブロックチェーンの仕組み

5.4.1 ブロックに含まれるデータ

ブロックにはハッシュ化した前の取引データの入ったブロック、取引データ、nonce が入っている。

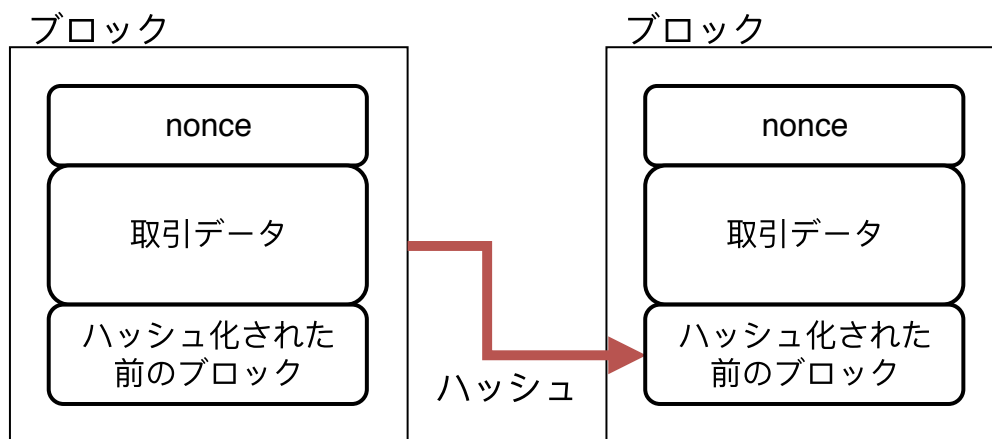


図 14 ブロックに含まれるデータ

5.4.2 nonce の計算

nonce は一定の長さの任意の値である。nonce に数字を入れていき、ハッシュ化されたブロックの頭に 0 が一定数続くときに nonce を確定する。

ハッシュの元データは復元することができないため、nonce は総当たりの的に探していくことになる。

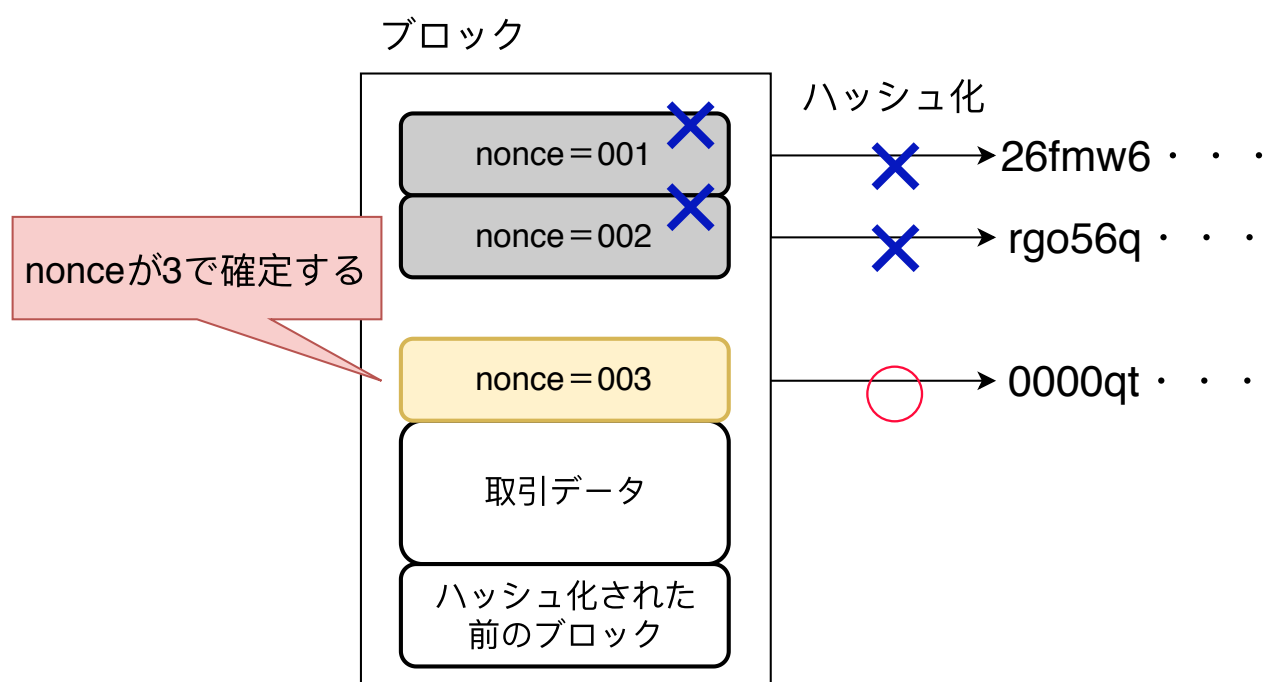


図 15 nonce の計算

そのためデータが改ざんされると、改ざんされたブロック以降の nonce を全て総当たりに探し直すこととなるためブロックチェーンが短くなる。ブロックチェーンが分岐したときは最も長いブロックチェーンのみを有効とするというルールがあるため、昔のブロックの改ざんほど難しくなる。

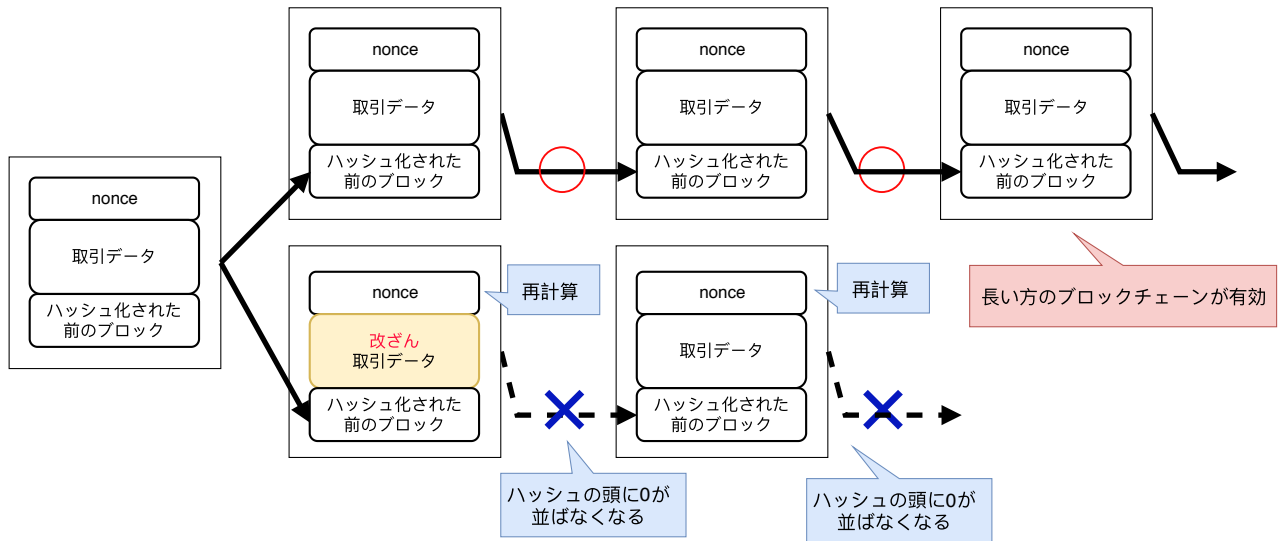


図 16 改ざんを防ぐ仕組み

6 本研究で提案する仮説

積み上げ型教科が苦手な学生は特定の箇所が苦手なのではなく、「どこがわかっていないのかもわからない」「そもそもいま何を教えられているのかもわからない」など根本的な部分で躓いていて、その原因として、前提知識には単元の学習目的が主単元で利用することとなっている単元が存在することだと考えた。このことから以下の問題が起こると考える。

- 学習の目的がわかりづらい

このような単元では問題を解くことが目的となっており、学習する単元の具体的なイメージができないことが問題であると考えた。そこで問題を解くことを目的にするのではなく、その技術を使って何ができるかを意識することが、理解の促進につながると考えた。

- どの単元の知識を使えばいいかがわかりづらい

複数の前提単元を用いることで、学習内容がどちらの単元のものであったかなど混乱する学生がいると考えた。そこで主単元の概要をあらかじめ学習し、複数の前提単元がどのように使われているかを明確にすることで、理解の促進につながると考えた。

このことから「主単元の概要をあらかじめ学習することで、理解を促進することができる」という仮説を立てた。

7 実験

本章では、提案した仮説が正しいことを証明するために実験を行う。以下で実験方法の解説を行う。

7.1 実験目的

本研究で提案した「主単元の概要をあらかじめ学習することで、前提単元の理解を促進することができる」という仮説の検証をすることを目的とする。

7.2 実験方法

7.2.1 実験期間

情報数学応用の講義にて3週かけて行う。

1週目：平成30年11月16日

2週目：平成30年11月30日

3週目：平成30年12月7日

7.2.2 実験方法

Aクラスでは仮説を用いない講義を行い、Bクラスでは仮説を用いた講義を行う。最後に両クラスで同一の小テストを行い、点数の比較と分析を行う。また小テストと同じ用紙にて比較対象を同条件にするためのアンケートを行う。

(1) 講義スケジュール

実験を行う講義は図17のように行う。仮説を用いないAクラスでは最初に暗号とハッシュについて学習した後に、ブロックチェーンについて学習する。仮説を用いたBクラスでは最初に暗号とハッシュを前提知識とするブロックチェーンの概要について学習した後に、暗号とは負について学習する。

	Aクラス	Bクラス
1週目	暗号の仕組み	ブロックチェーンとは
	ハッシュの仕組み	暗号の仕組み
2週目	ブロックチェーンとは	ハッシュの仕組み
	ブロックチェーンの仕組み	ブロックチェーンの仕組み
3週目	小テスト	小テスト

図17 講義スケジュール

(2) アンケートについて

アンケートにて「ブロックチェーンについて講義を受ける前に学習したことがあるか」について「はい」「いいえ」の二択で聞いた。「はい」と答えた学生は仮説の「あらかじめ内容を理解する」に該当するが、人数に偏りができることと、元の知識により高得点を取る可能性が高いことから、比較の対象とはせず参考程度に留める。

またアンケート未回答の学生も比較の対象としないこととする。

(3) 小テスト内容

小テストでは以下の内容を問う。

① RSA 暗号の特徴

② ハッシュと暗号の違い

③ ブロックチェーンの仕組み

点数配分は① + ②で5点、③で5点の合計10点とする。

(4) 評価方法について

「前提单元」「主单元」「合計」の3つの観点で点数を比較する。「前提单元」は ① RSA 暗号の特徴 ② ハッシュと暗号の違い、「主单元」は ③ ブロックチェーンの仕組み、「合計」は小テスト全体の点数（① + ② + ③）について平均点や偏差値を比較する。また元の学力の差を考慮して、中間試験の偏差値との差を比較する。

7.2.3 実験対象

千葉工業大学 情報科学部 情報ネットワーク学科の学生のうち

2018 年後期に開講された情報数学応用の受講者

A クラス 76 名, B クラス 76 名, 計 152 名とする。

ただし複数回の講義を用いての実験であるため、比較や分析に使用するデータは小テスト受験者のみである。またアンケートの結果、受講前にブロックチェーンを学習してことがない学生で、中間試験を受験している学生を比較と分析の対象とする。

8 実験結果

8.1 分析対象者に関するアンケートの結果

「ブロックチェーンについて講義を受ける前に学習したことがあるか」という質問の結果は表1のようになった。

表1 アンケート結果

	A クラス	B クラス
学習したことがない	60	56
学習したことがある	4	4
アンケート未回答	3	2
小テスト未受験	9	14

またブロックチェーンを学習したことがない学生のうち、中間試験を受験した学生は表2の通りとなった。

表2 中間試験の受験状況

	A クラス	B クラス
受験した	59	54
未受験	1	2

この結果より小テストの点数の比較はAクラスが59人、Bクラスが54人の計113人を対象として行う。

8.2 講義のわかりやすさに関するアンケートの結果

8.2.1 暗号

「暗号の講義はわかりやすかったか」という質問の結果は図 18 のようになった。

両クラス共にわかりやすいと答えた学生が約 60 % となり，とてもわかりやすいと答えた学生と合わせて約 80 % という結果となった。

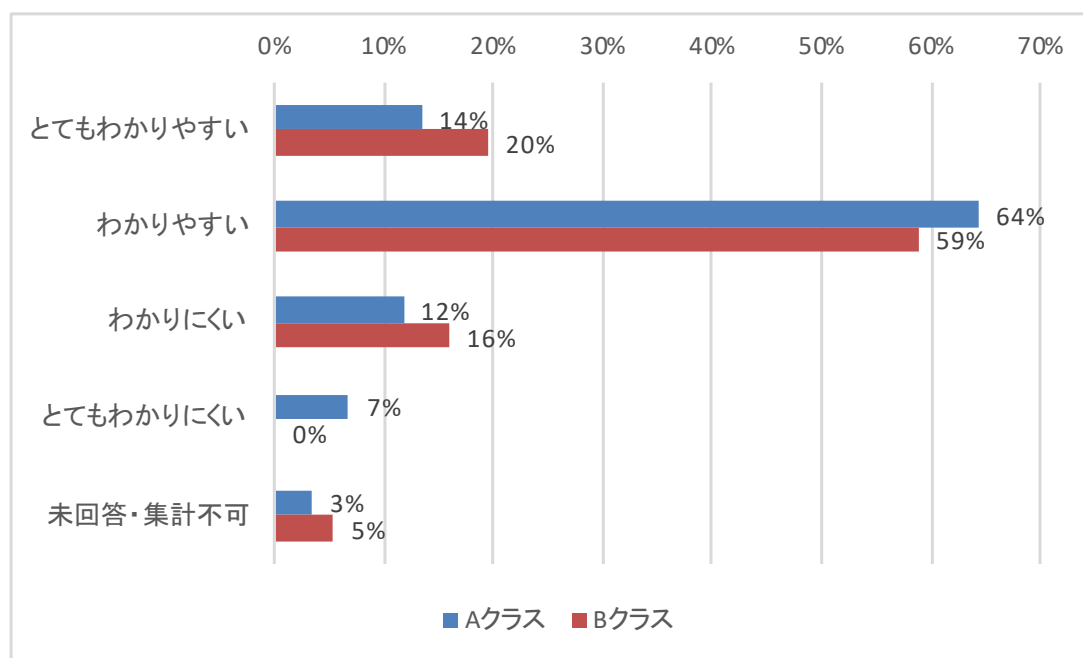


図 18 暗号の講義についてのアンケート結果

8.2.2 ハッシュ

「ハッシュの講義はわかりやすかったか」という質問の結果は図 19 のようになった。

両クラス共に「わかりにくい」と答えた学生の割合が一番高く、「わかりやすい」と答えた学生の割合が次に高いという結果になった。また「とてもわかりにくい」と答えた学生は A クラスでは 2 %，B クラスでは 0 % となり，暗号の講義と比べ両クラス共にわかりにくいと感じる学生が多いという結果になった。

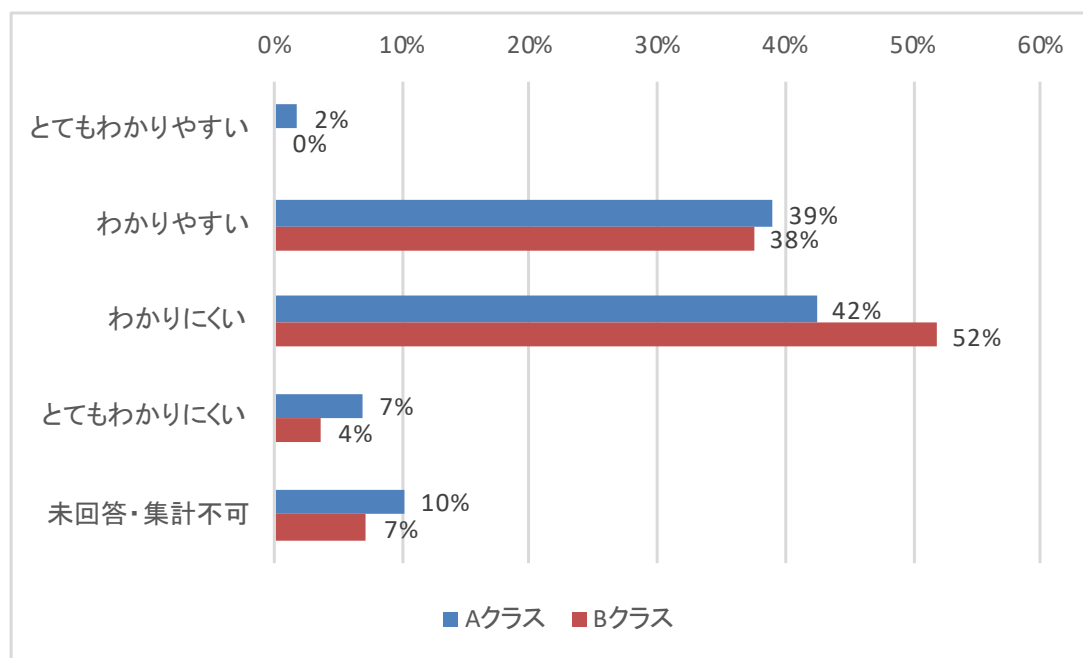


図 19 ハッシュの講義についてのアンケート結果

8.2.3 ブロックチェーン

「ブロックチェーンの講義はわかりやすかったか」という質問の結果は図 20 のようになった。

両クラス共に「わかりにくい」と答えた学生の割合が一番高く、「わかりやすい」と答えた学生の割合が次に高くなり、ハッシュの講義に近い結果となった。

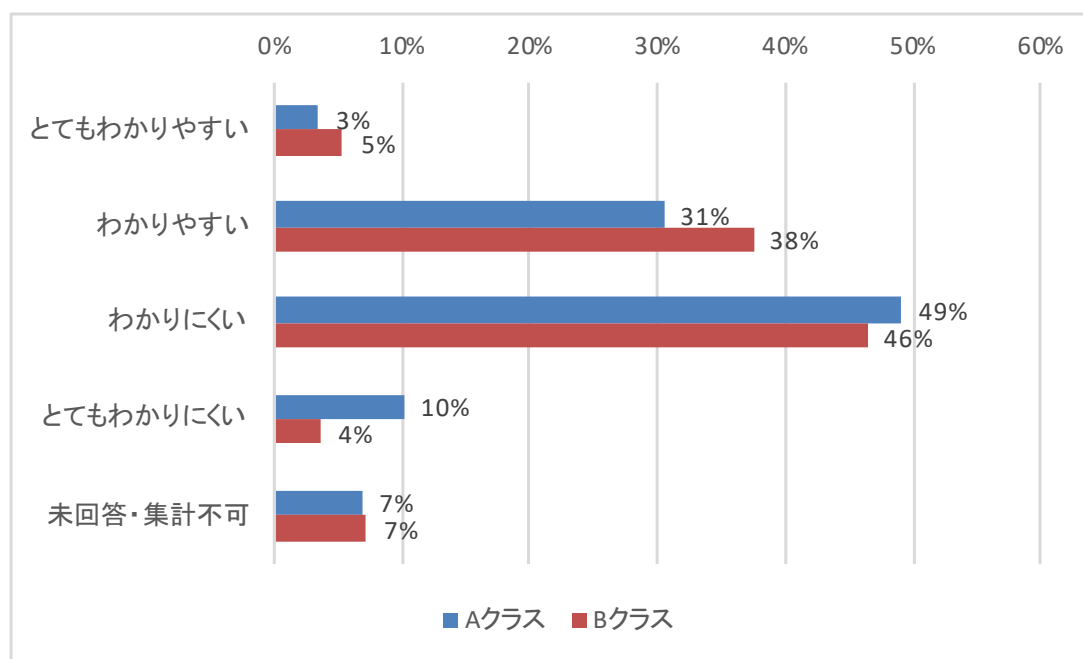


図 20 ブロックチェーンの講義についてのアンケート結果

8.3 小テストの点数

8.3.1 前提单元

前提单元は5点満点で、点数分布は図21のようになった。Aクラスは4点の学生が一番多く約40%で、点数が低くなるにつれて割合が低くなった。Bクラスは3点の学生が一番多く約30%で、Aクラスと同様に点数が低くなるにつれて割合が低くなったが、Aクラスよりも5点の学生の割合はわずかに高かった。また両クラス共に0点の学生はいなかった。

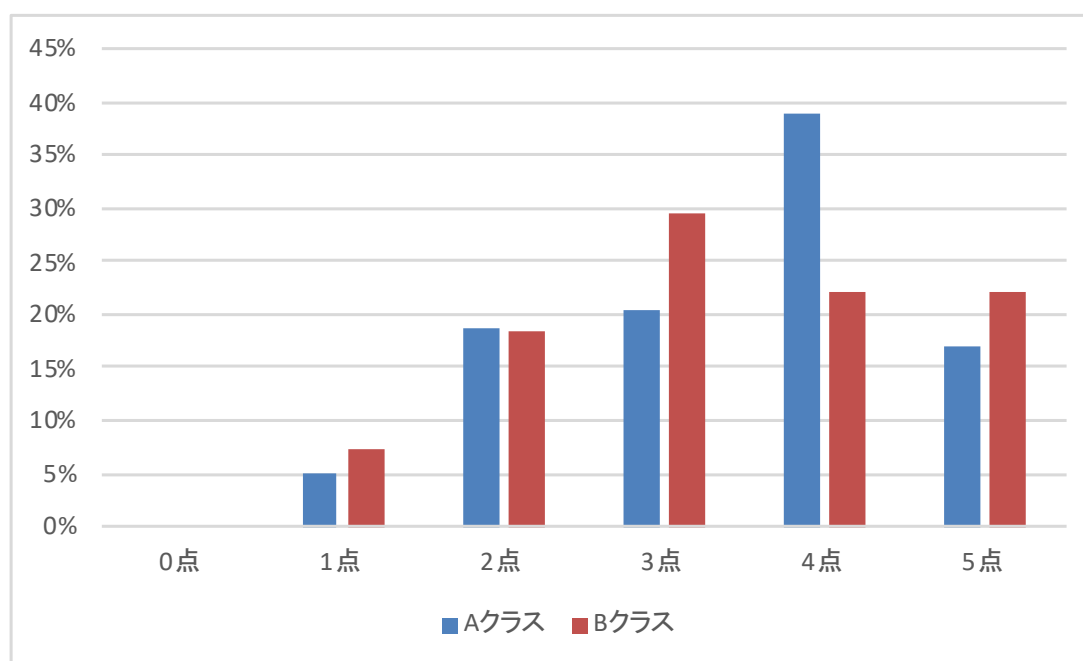


図 21 前提单元の点数分布

また前提单元の各クラスの偏差値と平均点は表3のようになった。平均点はAクラスの方がわずかに高くなったが、中間試験の偏差値と比較したところほとんど差は見られなかった。

表 3 前提单元の偏差値と平均点

	A クラス		B クラス	
	偏差値	平均点	偏差値	平均点
前提单元	50.4	3.4 /5	49.5	3.3 /5
中間試験	50.6	29.4 /50	49.3	28.0 /50

8.3.2 主単元

主単元は 5 点満点で、点数分布は図 22 のようになった。両クラス共に 3 点の学生が一番多く、4 点以上の学生は人数が少なくなった。A クラスはまた両クラス共に 0 点の学生はいなかった。

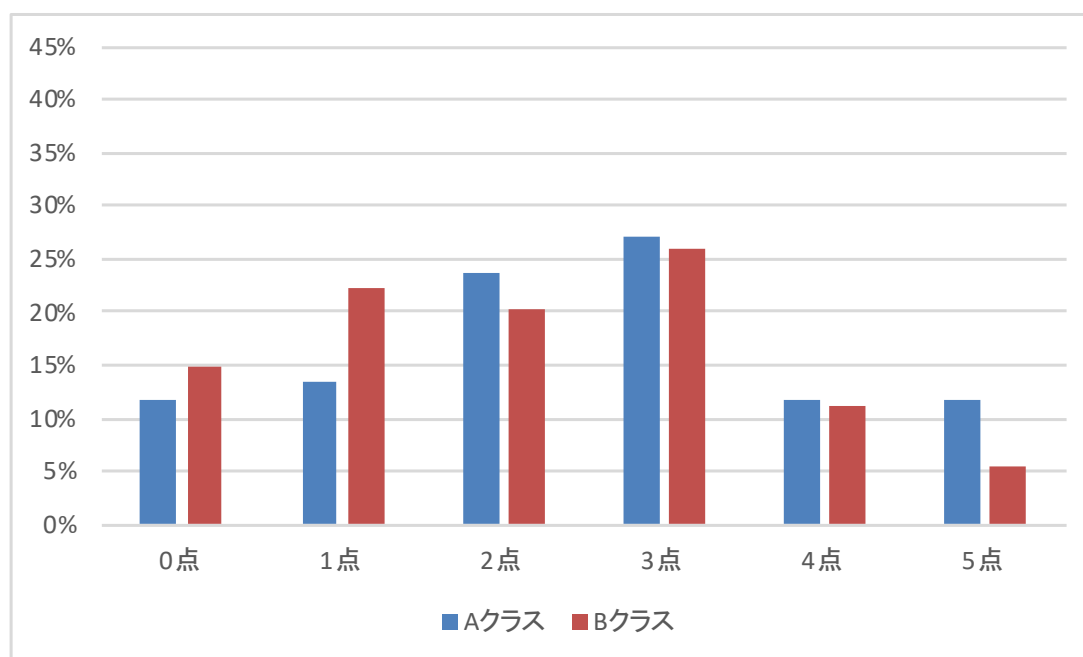


図 22 主単元の点数分布

また主単元の各クラスの偏差値と平均点は表 4 のようになった。平均点は A クラスの方がわずかに高くなった。また中間試験の偏差値と比較しても A クラスの方が高くなった。

表 4 主単元の偏差値と平均点

	A クラス		B クラス	
	偏差値	平均点	偏差値	平均点
主単元	51.2	2.5 /5	48.7	2.1 /5
中間試験	50.6	29.4 /50	49.3	28.0 /50

8.3.3 小テスト全体

小テスト全体の点数は 10 点満点で、点数分布は図 23 のようになった。両クラス共に 6 点の学生が一番多く、点数が高くなるにつれて人数が少なくなった。A クラスは 7 点以上の割合が全体的に高く、B クラスは 4 点以下の割合が全体的に高くなった。

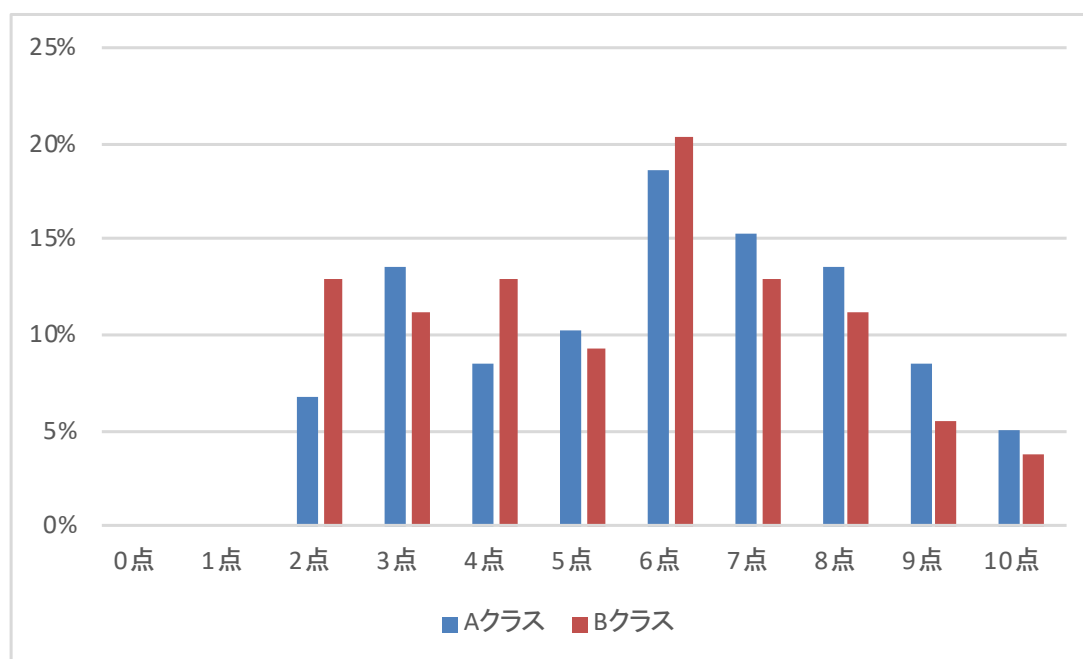


図 23 小テスト全体の点数分布

小テスト全体の偏差値と平均点は表 5 のようになった。平均点は A クラスの方がわずかに高くなった。また中間試験の偏差値と比較しても A クラスの方が高くなった。

表 5 小テストの偏差値と平均点

	A クラス		B クラス	
	偏差値	平均点	偏差値	平均点
小テスト	51.0	5.9 /10	48.9	5.6 /10
中間試験	50.6	29.4 /50	49.3	28.0 /50

9 考察

今回の実験では仮説を証明することができなかった。そこで実験方法に関する原因や改善点を以下のように考察した。

9.1 スケジュール

まず今回の実験は講義にて行った。一週目と二週目の間に休講日があり、時間が空いてしまったことから、講義外での勉強量が深く関係してしまい、講義での理解度が測ることができなかった可能性がある。そのため講義ではなく別の方法を用いて、期間を開けずに小テストまで終わらせることや、講義外での自習を行えない環境にするなど、別の実験方法で行うべきである。

また講義と小テストを別日で行ったため講義を欠席して小テストのみ受けた学生などの考慮が甘く、講義と小テストを受けた学生のみで実験を行うべきであった。

9.2 試験問題

図6で示すように「前提单元」「主单元」の問題の難易度に差ができてしまった。簡単な問題は講義から時間が経っていても解けるが、難解な問題は内容を忘れやすく、前述した休講日も合間って時間が経っているクラスの点数に影響が出ている可能性がある。そのため前提单元と主单元の問題難易度を合わせる必要がある。

表6 前提知識を用いた单元の平均点の比較

	A クラス	B クラス
前提单元	3.44	3.33
主单元	2.49	2.13

9.3 授業の選定

今回の実験で前提单元として扱った暗号やハッシュには他の講義で扱った行列や誤り検出訂正などの前提知識が必要である。元の学力の差を考慮するために中間試験の点数との比較を行ったが、本実験に関係する知識の元の学力を測るものとは言えなかったため点数にズレが生じてしまった可能性がある。

また仮説を提唱した理由の一つが「前提单元の目的をはっきりさせることで理解が促進できる」ことであったが、今回の実験で用いた暗号技術は実用例が割と身近にあり、ブロックチェーンを目的にするよりもわかりやすいと感じる学生が多かったと考えた。

10 結言

各塾の調査では中学、高校で学習する科目の中で数学と英語は苦手になりやすいと言われている [1]. この 2 科目の共通点である積み上げ型教科という点に注目した. 積み上げ型教科とは既に学習した知識を使うことを前提として授業を行う教科である. 本研究では前提知識となる単元を「前提単元」、前提知識を用いて学習を行う単元を「主単元」と称した. 前提単元には学習目的が主単元で利用することになっているものがあり, このような単元は学習目的や実用例などのイメージがしづらく, 理解の妨げになっていると考えた. また前提単元が理解できないと主単元にまで影響を及ぼしてしまい, 理解できない単元が増えるため, 苦手な教科になってしまう.

そこで本研究では積み上げ型教科での理解力向上を目的として, 「学習する単元を前提知識とする単元の概要をあらかじめ説明することで, 内容の理解を促進することができる」という仮説を立て, 仮説の証明を目指した. しかし今回行った実験のみでは仮説が正しいことを証明することができなかった. しかし実験方法に多くの改善点が見つかったため, 仮説が間違っていると断定することもできないという結果になった.

11 謝辞

本研究の遂行および本論文の作成にあたり，多くの御助言と御指導いただきました須田宇宙准教授に深く感謝の意を表します．また実験に協力していただいた情報数学応用の受講者に深く感謝の意を表します．

参考文献

- [1] ベネッセ教育情報サイト: “教科学習が不得意と感じている高校生が9割！ そのほとんどが英語と数学に偏るのにはある理由が”, <https://www.benesse.jp/kyouiku/201603/20160329-3.html>, (参照 2018-8-14)
- [2] Satoshi Nakamoto, “Bitcoin:A Peer-to-Peer Electronic Cash System” ,2009 年
- [3] ” コンセンサスアルゴリズムとは？ 「PoW・PoS・PoI」 の種類と違いを解説”. はじめてのビットコイン. <https://www.newscrypto.jp/articles/2612>, (参照 2019-1-9)