

Использование технологии блокчейн для защиты данных

Д.В. Балдов, С.Ю. Петрова, А.А. Лебедев

Аннотация – В настоящее время ценность информации практически приравнялась к ценности материальных объектов. Современные отрасли оперируют огромными потоками данных, к которым предъявляются повышенные требования сохранности и достоверности. Так и Нижегородский государственный инженерно-экономический университет (НГИЭУ) хранит данные о дипломах, выданных своим выпускникам. Информация о каждом выданном дипломе не подлежит изменению или исправлению.

В университете обучается по основным профессиональным образовательным программам высшего образования – программам аспирантуры, программам магистратуры, программам бакалавриата, а также по программам подготовки специалистов среднего звена, программам профессионального обучения, дополнительным профессиональным образовательным программам более 5 800 человек. [1]

Ежегодно требуется добавление новых данных о выданных дипломах и их надежное хранение. Защита от несанкционированного доступа выходит на первый план. Недопустимо чтобы какая-либо информация была изменена или удалена. Разработанное клиент-серверное приложение позволит хранить в базе данных вуза информацию о выданных дипломах. А использование шифрования и технологии блокчейн в должной мере защитит данные от порчи или их изменения третьими лицами. Для решения поставленной задачи хорошо подходит технология блокчейн. Данная технология за счет своей простоты позволит защитить хранимые данные.

В статье рассматривается создание клиент-серверного приложения, которое каждую строку записи о выданном дипломе удостоверяет специальной записью с хеш-суммой. Из строк данных выстраиваются цепочки, которые невозможно исправить или удалить. В случае если произошла подмена данных или их удаление, то система сможет определить исправленную строку и в автоматическом режиме запросить испорченные данные с других серверов. В итоге полностью исключается потеря или порча данных.

Разработанное приложение и база данных позволили продемонстрировать возможности технологии блокчейн. Технология имеет достаточно большой потенциал для применения ее в любых сферах деятельности, где требуется защита информации от ее изменения или удаления. Разработанное программное обеспечение показало свою эффективность и будет внедряться в учебный процесс. Что в значительной степени может

повысить защищенность информации к ее изменению или удалению.

Ключевые слова: блокчейн, базы данных, диплом, защита данных, криптозащита, несанкционированный доступ, связные списки, СУБД, цепочка, шифрование.

I. Введение

Блокчейн (англ. blockchain) — выстроенная по определённым правилам непрерывная последовательная цепочка блоков (связный список), содержащих информацию. Чаще всего копии цепочек блоков хранятся на множестве разных компьютеров независимо друг от друга. [2]

Сегодня тема блокчейна находится на пике моды. Любой проект вызывает повышенный интерес, если он «на блокчейне», любая газетная статья привлекает к себе дополнительное внимание, если в ней упомянута эта технология. [3]

Основой технологии является использование связанных списков. Связанный список представляет собой организацию данных, при которой каждый блок данных (за возможным исключением первого и последнего) ссылаются на один-два других. Это является основным отличием блокчейна от большинства современных хранилищ данных, так как в них используются преимущественно массивы.

Массив представляет собой совокупность переменных одного типа с общим для обращения к ним именем. В C# массивы могут быть как одномерными, так и многомерными. Массивы служат самым разным целям, поскольку они предоставляют удобные средства для объединения связанных вместе переменных. [4]

Принципиальным преимуществом перед массивом является структурная гибкость: порядок элементов связанного списка может не совпадать с порядком расположения элементов данных в памяти компьютера, а порядок обхода списка всегда явно задаётся его внутренними связями. [5]

Блокчейн является расширением связанных списков с одним основным ограничением: связи между блоками данных и сами данные являются неизменными. Наиболее простая аналогия – стопка книг.

- Книги лежат одна на другой;
- добавить новую книгу, не развалив стопку, можно только наверх;
- достать книгу из стопки нельзя, иначе она развалится.

По сути, блокчейн — это новая организационная парадигма для координации любого вида человеческой деятельности. Возможно даже, что это наше будущее, о котором полезно узнать уже сегодня. [6]

Неизменность связей между блоками данных обеспечивается за счет привязки не только к блоку, но и к его данным. Данные каждого блока подаются на вход хэш-функции. Результатом ее выполнения является числовая последовательность фиксированной длины.

В общем случае (согласно принципу Дирихле) нет однозначного соответствия между исходными (входными) данными и хеш-кодом (выходными данными). Возвращаемые хеш-функцией значения (выходные данные) менее разнообразны, чем значения входного массива (входные данные). Случай, при котором хеш-функция преобразует несколько разных сообщений в одинаковые сводки, называется «коллизией». [7]

Хеширование применяется для построения ассоциативных массивов, поиска дубликатов в сериях наборов данных, построения достаточно уникальных идентификаторов для наборов данных, контрольное суммирование с целью обнаружения случайных или намеренных ошибок при хранении или передаче. [8]

При изменении данных меняется выход хэш-функции (для примера взята SHA256), что позволяет обеспечить сохранность данных за счет возможности отследить их изменения. Шифрование — обратимое преобразование информации в целях сокрытия от неавторизованных лиц, с предоставлением, в это же время, авторизованным пользователям доступа к ней. Главным образом, шифрование служит задачей соблюдения конфиденциальности передаваемой информации. Важной особенностью любого алгоритма шифрования является использование ключа, который утверждает выбор конкретного преобразования из совокупности возможных для данного алгоритма. [9]

Важно понимать, что хеширование не синоним шифрования. Шифр создается для того, чтобы по нему можно было восстановить исходный набор данных. Но для хэша это условие необязательно, а в нашем случае, даже нежелательно.

$\text{Hash} = \text{SHA256}(\text{Data});$

Неизменность связей между блоками обеспечивается за счет того, что каждый последующий блок при хешировании своих данных использует данные предыдущего блока, например их хэш. Наиболее частой причиной такого является то, что одному хэшу соответствует бесконечное множество входных данных (рис. 1).

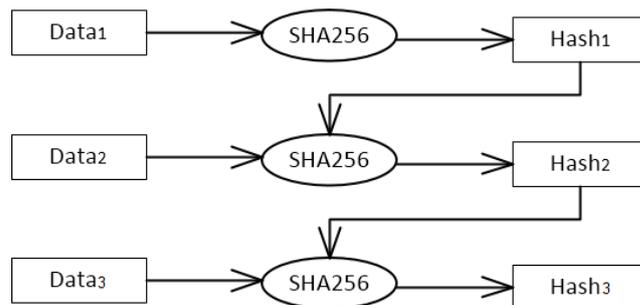


Рисунок 1 – Алгоритм хеширования

Очень полезной является возможность представить цепочку блоков в виде таблицы. Это позволяет использовать для хранения реляционные базы данных, а значит применять все существующие способы шифрования, сжатия и защиты данных, применимые для этих баз данных.

II. Применение технологии

На данный момент технология широко применяется в сфере рынка криптовалют. Наибольшую известность технологии принесло использование в платежной платформе Bitcoin. Однако данная платформа использует не только блокчейн, но и множество других современных и перспективных технологий, таких как технология цифровой подписи.

В настоящее время уже существует ряд расширений для разработки бизнес-приложений на блокчейн, обеспечивающих:

- безопасное администрирование сетей, исключая хакерские атаки МИМ («человек посередине») и снимающее проблему «единого администратора»;
- хранение цифровых сертификатов, делающее полностью защищенным доступ пользователей к сайтам (в частности, исключая перехват паролей);
- безопасные двусторонние сделки без привлечения гарантирующей третьей стороны (юридической фирмы, нотариуса, банка и др.);
- фиксацию времени размещения документов, позволяющую решать вопросы патентования, авторского права и др.;
- подтверждение подлинности продукта (товара) с помощью надежно защищенного сертификата;
- подтверждение прав на любую собственность;
- создание общедоступных электронных визиток, информация на которых автоматически обновляется даже после «раздачи» по интернет-ресурсам;
- систему DNS, неуязвимую для DDOS-атак. [10]

Биткойн — это набор концептов и технологий, которые совместно образуют основу для экосистемы цифровых денег. Денежные единицы, называемые биткойнами, используются для сохранения и передачи стоимости между участниками сети. [11]

Биткойн, возможно, является одним из крупнейших событий в области финансов с момента появления валюты. With Understanding Bitcoin, эксперт-автор Педро Франко предоставляет специалистам в области

финансов полный технический справочник и ресурс для криптографии, инженерного и экономического развития биткойнов и других криптотермин. Эта всеобъемлющая, но доступная работа полностью исследует поддерживающие экономические реалии и технологические достижения Биткойна и представляет положительные и отрицательные аргументы различных экономических школ относительно ее постоянной жизнеспособности. [12]

Переводы средств в одноранговой сети от отправителя к получателю осуществляются без участия третьих лиц. Это позволяет избежать эффекта двойной оплаты и полностью исключает авторитетность любого пользователя. [13]

III. Массачусетский технологический институт

Массачусетский технологический институт (MIT) использовал блокчейн Биткойна для выпуска цифровых дипломов более чем 100 выпускникам. По словам соучредителя и генерального директора LearningMachine Криса Джегерса (ChrisJagers), MIT входит в число первых университетов, выпустивших «официальные записи в формате, который может существовать, даже если учреждения не станут» [14].

Цель проекта – создать децентрализованную платформу, позволяющую выпускникам учебных заведений безопасно хранить и делиться данными о своих дипломах, что по задумке авторов проекта должно упростить дальнейшее трудоустройство. Предполагается, что система будет полностью децентрализованная, что должно обеспечить сохранность и защиту данных, но при таком подходе могут возникнуть проблемы достоверности хранимых данных.

Хотя сама информация о дипломе не хранится в цепочке блоков, система использует временную транзакцию, указывающую, что MIT создал цифровую запись для сертификата. Это позволяет студенту подтвердить свое право на получение диплома позднее. [15]

IV. Описание программы

На данный момент на базе нашего университета ведется разработка программного продукта с использованием технологии блокчейн. Его предназначение: сохранение данных о выданных дипломах. Уже разработан прототип программы. Его функционал: создание и добавление новой записи в конец цепочки и проверка таблицы на согласованность. Также на время разработки для проверки согласованности были добавлены элементы, позволяющие изменять записи и пересчитывать их хэш-суммы. В качестве языка программирования был взят язык C#.

C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык имеет статическую типизацию,

поддерживает полиморфизм, перегрузку операторов (в том числе операторов явного и неявного приведения типа), делегаты, атрибуты, события, свойства, обобщенные типы и методы, итераторы, анонимные функции с поддержкой замыканий, LINQ, исключения, комментарии в формате XML. [16].

Но главный критерий выбора C# обусловлен упрощенной работой с графическим интерфейсом. В то время, как в C++ CLR код для формы неизбежно перегружается сложными конструкциями из дескрипторов, переходов по указателям и явным приведений типов.

Для создания графических интерфейсов с помощью платформы .NET применяются разные технологии - WindowsForms, WPF, приложения для магазина WindowsStore (для ОС Windows 8/8.1/10). Однако наиболее простой и удобной платформой до сих пор остается WindowForms или формы. [17]

Создание новой записи

Данные хранятся в базе данных SQL. Создание записи начинается с заполнения формы-конструктора (рис. 2).

Рисунок 2 – Ввод данных в базу данных

Формы являются основными строительными блоками. Они предоставляют контейнер для различных элементов управления. А механизм событий позволяет элементам формы отзываться на ввод пользователя, и, таким образом, взаимодействовать с пользователем. [18]

После окончания ввода значения полей формы форматируются в строку и проходят через алгоритм хеширования. Выходное значение записывается вместе с данными в базу данных. Если это не первая запись, то к строке, сформированной из данных формы, добавляется хэш предыдущей записи. Таким путем происходит заполнение базы данных.

Список представляет собой последовательность элементов, каждый из которых содержит значение некоторого типа и адрес следующего элемента (причем для последнего из них адрес может быть нулевым). К любой такой последовательности всегда можно

добавить еще один элемент (хотя реальная попытка подобного добавления может закончиться неудачно, если отведенная программе свободная память исчерпана). [19]

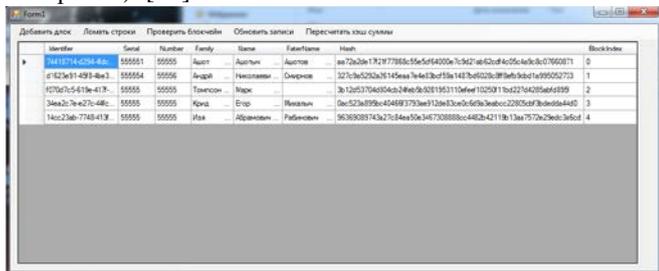


Рисунок 2 – Главная форма приложения

Данные отображаются в главной форме приложения с помощью элемента DataGridView. Элемент управления DataGridView, может отображать и изменять табличные данные из различных типов источников данных. Привязка данных к DataGridView — это легко-настраиваемое и интуитивно понятное свойство DataSource. [20]

V. Проверка таблицы на согласованность

Проверка корректности цепочки начинается с первого блока. Его данные форматируются в одну строку и подаются на вход алгоритма хэширования. Если выходное значение отличается от изначально полученного хэша, то выводится соответствующее сообщение. Далее проверяется второй блок. Для него и последующих блоков последовательность проверки точно такая же, за исключением того, что при формировании строки из данных текущего блока в ее конец добавляется хэш предыдущего блока. Если после проверки всех записей ошибок не найдено, то программа выводит сообщение об этом.

Выдача сообщений происходит через вызов функции Show() класса MessageBox.

Класс MessageBox отображает окно сообщения (диалоговое окно) с текстом для пользователя. Это модальное окно, блокирующее другие действия в приложении, пока пользователь не закроет его. MessageBox может содержать текст, кнопки и символы, с помощью которых информируется и инструктируется пользователь. [21]

Хэш-функция

Для хэширования используется функция, использующая алгоритм хеширования SHA256. SHA256 принадлежит к функциям семейства SHA2.

SHA-2 (англ. Secure Hash Algorithm Version 2 — безопасный алгоритм хеширования, версия 2) — семейство криптографических алгоритмов — однонаправленных хэш-функций, включающее в себя алгоритмы SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/256 и SHA-512/224.[22]

Первоначальная версия алгоритма SHA-256 была создана Агентством национальной безопасности США весной 2002 года. Спустя несколько месяцев

Национальный метрологический университет опубликовал новоявленный протокол шифрования в принятом на федеральном уровне стандарте безопасной обработки данных FIPS PUB 180-2.[23]

В нашей программе после того, как хэш от данных получен он сам подается на вход SHA256. Таким образом мы получаем хэш от хэша. Данную операцию можно повторять достаточно много раз благодаря обеспечиваемой скорости хэширования. Делается для того, чтобы по хэшу невозможно было получить исходные данные.

Программно получилось очень удобно реализовать алгоритм хэширования как функцию отдельного модуля(класса).

Функция являет собой небольшую подпрограмму. Если просто программа — это решение какой-то прикладной задачи, то функция — это тоже решение, только уже в рамках программы и, соответственно, она выполняет задачу «попроще». Функции позволяют уменьшить размер программы за счет того, что не нужно повторно писать какой-то фрагмент кода - мы просто вызываем сколько угодно и где нужно объявленную функцию. [24]

Класс представляет собой шаблон, по которому определяется форма объекта. В нем указываются данные и код, который будет оперировать этими данными. [25]

Благодаря такой реализации удалось избежать копирования кода для обработки создания записи и функции проверки блокчейна на наличие ошибок. Единственное отличие этих операций в том, что во втором случае функция обрамлена оператором while.

Оператор while выполняет оператор или блок операторов, пока определенное выражение не примет значение false. [26]

VI. Заключение

На данный момент готова лишь демонстрационная версия программы. В финальную версию планируется внести ряд добавлений: возможность создания пароля, реализовать резервное копирование, распределенное хранение данных (например, по примеру криптовалют, либо централизованно) и шифрование данных.

Внедрение и апробация разрабатываемой ИС в систему выдачи и хранения данных о выданных дипломах ГБОУ ВО НГИЭУ.

Также данный программный продукт можно адаптировать под другие виды контента без внесения изменений в логику программы.

Основным преимуществом программы является повышенная защищенность данных от несанкционированного изменения. Для того чтобы изменить одну запись необходимо пересчитать ее хэш и хэши последующих записей.

Из недостатков следует выделить: несколько большие затраты времени на добавление новой записи и отсутствие кроссплатформенности.

БИБЛИОГРАФИЯ

- [1] Публичный отчёт ректора НГИЭУ об итогах работы в 2016 – 2017 // Нижегородский государственный инженерно-экономический университет URL: http://ngiei.ru/public_rektor (дата обращения: 26.06.18).
- [2] Блокчейн // wikipedia.org URL: <https://ru.wikipedia.org/wiki/Блокчейн> (дата обращения: 26.06.18).
- [3] Артем Генкин, Алексей Михеев. Блокчейн. Как это работает и что ждет нас завтра. — М.: Альпина Паблишер, 2017. — 281с.
- [4] Массивы // professorweb.ru URL: https://professorweb.ru/my/csharp/charp_theory/level_4/4_1.php (дата обращения: 26.06.18).
- [5] Связный список // wikipedia.org URL: https://ru.wikipedia.org/wiki/Связный_список (дата обращения: 26.06.18).
- [6] Мелани Свон. Блокчейн: Схема новой экономики. — Олимп-Бизнес, 2016. — 240 с.
- [7] Хеширование // wikipedia.org URL: <https://ru.wikipedia.org/wiki/Хеширование> (дата обращения: 26.06.18).
- [8] Хеширование // academic.ru URL: <https://dvc.academic.ru/dic.nsf/ruwiki/78851> (дата обращения: 26.06.18).
- [9] Шифрование // wikipedia.org URL: <https://ru.wikipedia.org/wiki/Шифрование> (дата обращения: 26.06.18).
- [10] Что такое Blockchain (блокчейн)? Технология, платформа, транзакции. // mining-cryptocurrency URL: <https://mining-cryptocurrency.ru/blockchain/> (дата обращения: 26.06.18).
- [11] Andreas M. Antonopoulos. 7. The Blockchain // Mastering Bitcoin. — O'Reilly Media, Inc., 2014.
- [12] Pedro Franco. The Blockchain // Understanding Bitcoin: Cryptography, Engineering and Economics. — John Wiley & Sons, 2014.
- [13] Bitcoin: A Peer-to-Peer Electronic Cash System // Биткойн URL: <https://bitcoin.org/bitcoin.pdf> (дата обращения: 22.06.2018).
- [14] MIT выпустил 100 дипломов на блокчейне // bits.media URL: <https://bits.media/news/mit-vypustil-100-diplomov-na-blokcheyne/> (дата обращения: 26.06.18).
- [15] 100 Diplomas: MIT Issues Graduate Certificates on a Blockchain App // coindesk.com URL: <https://www.coindesk.com/100-diplomas-mit-issues-graduate-certificates-on-a-blockchain-app/> (дата обращения: 26.06.18).
- [16] C Sharp // wikipedia.org URL: https://ru.wikipedia.org/wiki/C_Sharp (дата обращения: 26.06.18).
- [17] Введение в Windows Forms // metanit.com URL: <https://metanit.com/sharp/windowsforms/1.1.php> (дата обращения: 26.06.18).
- [18] Работа с формами // metanit.com URL: <https://metanit.com/sharp/windowsforms/2.1.php> (дата обращения: 26.06.18).
- [19] Пример связанного списка // C++ для начинающих URL: http://mycpp.ru/cpp/book/c05_11.html (дата обращения: 26.06.18).
- [20] Общие сведения об элементе управления DataGridView (Windows Forms) // microsoft.com URL: <https://docs.microsoft.com/ru-ru/dotnet/framework/winforms/controls/datagridview-control-overview-windows-forms> (дата обращения: 26.06.18).
- [21] Класс MessageBox // msdn.microsoft.com URL: [https://msdn.microsoft.com/ru-ru/library/system.windows.forms.messagebox\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/system.windows.forms.messagebox(v=vs.110).aspx) (дата обращения: 26.06.18).
- [22] SHA2 // https://ru.wikipedia.org/wiki/SHA-2 (дата обращения: 26.06.18).
- [23] SHA-256 – живая классика среди алгоритмов хеширования // tutdenegki.com URL: <http://tutdenegki.com/sha-256> (дата обращения: 26.06.18).
- [24] Функции в Си-шарп. Оператор return // mycsharp.ru URL: http://mycsharp.ru/post/13/2013_05_21_funkcii_v_si-sharp_operator_return.html (дата обращения: 26.06.18).
- [25] Классы // professorweb.ru URL: https://professorweb.ru/my/csharp/charp_theory/level_5/5_1.php (дата обращения: 26.06.18).
- [26] while (Справочник по C#) // microsoft.com URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/while> (дата обращения: 26.06.18).

Балдов Дмитрий Валентинович родился в п.Тумботино Павловского района в 1985 году. Окончил в 2004 году ГБПОУ «Павловский автомеханический техникум имени И.И.Лепсе» по специальности «Техническая эксплуатация и обслуживание электрического и электромеханического оборудования», в 2010 году Нижегородский государственный технический университет им. Р.Е. Алексеева по специальности «Информационные системы и технологии». В 2020 году защитил диссертацию на соискание степени кандидата экономических наук. С 2013 года является преподавателем кафедры «Информационные системы и технологии» Нижегородского государственного инженерно-экономического университета, Княгинино, Россия. Текущие научные интересы включают: разработка программного обеспечения и баз данных, компьютерная графика, механика. E-mail: dimon170@rambler.ru

Петрова Светлана Юрьевна, родилась в Чувашской АССР, в 1988 году. В 2010 году с отличием окончила ГОУ ВПО "Чувашский государственный педагогический университет им. И.Я. Яковлева" по специальности "Математика" с дополнительной специальностью "Информатика". С 2011 года работает преподавателем кафедры "Информационные системы и технологии" ГБОУ ВО "НГИЭУ" (Княгининский университет). Степень кандидата экономических наук защитила в 2013 году в Княгининском университете. Ученое звание доцента по специальности "Системы автоматизации проектирования (по

отраслям)" присвоена в 2019 году. Научные интересы: проектирование и моделирование информационных систем, цифровая экономика.
E-mail: svet27ik@mail.ru

Лебедев Александр Анатольевич. В 2015 году закончил бакалавриат в Нижегородском государственном инженерно-экономическом университете по направлению подготовки 09.03.02 информационные системы и технологии

Using blockchain technology for data protection

D.V. Baldov, S.Yu. Petrova, A.A. Lebedev

Abstract – At present, the value of information is practically equal to the value of material objects. Modern industries operate with huge data streams, which are subject to increased security and reliability requirements. Likewise, the Nizhny Novgorod State Engineering and Economic University (NGIEU) stores data on diplomas issued to its graduates. Information about each issued diploma is not subject to change or correction.

At the university, more than 5,800 people study in the basic professional educational programs of higher education - postgraduate programs, master's programs, bachelor's programs, as well as training programs for mid-level specialists, vocational training programs, additional professional educational programs.

The addition of new data on issued diplomas and their reliable storage is required annually. Protection against unauthorized access comes to the fore. It is unacceptable that any information has been changed or deleted. The developed client-server application will allow storing information on the issued diplomas in the university database. And the use of encryption and blockchain technology will adequately protect data from damage or alteration by third parties. Blockchain technology is well suited for solving this problem. This technology, due to its simplicity, will protect the stored data.

The article discusses the creation of a client-server application, which certifies each line of the record about the issued diploma with a special record with a hash-sum. Chains are built from rows of data that cannot be corrected or deleted. If there was a substitution of data or their deletion, the system will be able to determine the corrected string and automatically request the corrupted data from other servers. As a result, data loss or damage is completely excluded.

The developed application and database made it possible to demonstrate the capabilities of blockchain technology. The technology has a sufficiently large potential for its application in any areas of activity where information protection is required from its modification or deletion. The developed software has shown its effectiveness and will be introduced into the educational process. That can greatly increase the security of information to its change or deletion.

Key words: blockchain, databases, diploma, data protection, crypto protection, unauthorized access, linked lists, DBMS, chain, encryption.

REFERENCES

- [1] Publichnyy otchet rektora NGIEU ob itogakh raboty v 2016 – 2017 // Nizhegorodskiy gosudarstvennyy inzhenerno-ekonomicheskii universitet URL: http://ngiei.ru/public_rektor (data obrashcheniya: 26.06.18).
- [2] Blokcheyn // wikipedia.org URL: <https://ru.wikipedia.org/wiki/Blokcheyn> (data obrashcheniya: 26.06.18).
- [3] Artem Genkin, Aleksey Mikheyev. Blokcheyn. Kak eto rabotayet i chto zhdet nas zavtra. — M.: Al'pina Pabliisher, 2017. — 281c.
- [4] Massivy // professorweb.ru URL: https://professorweb.ru/my/csharp/charp_theory/level/4/4_1.php (data obrashcheniya: 26.06.18).
- [5] Svyaznyy spisok // wikipedia.org URL: https://ru.wikipedia.org/wiki/Svyaznyy_spisok (data obrashcheniya: 26.06.18).
- [6] Melani Svon. Blokcheyn: Skhema novoy ekonomiki. — Olimp-Biznes, 2016. — 240 s.
- [7] Kheshirovaniye // wikipedia.org URL: <https://ru.wikipedia.org/wiki/Kheshirovaniye> (data obrashcheniya: 26.06.18).
- [8] Kheshirovaniye // academic.ru URL: <https://dvc.academic.ru/dic.nsf/ruwiki/78851> (data obrashcheniya: 26.06.18).
- [9] Shifrovaniye // wikipedia.org URL: <https://ru.wikipedia.org/wiki/Shifrovaniye> (data obrashcheniya: 26.06.18).
- [10] Chto takoye Blockchain (blokcheyn)? Tekhnologiya, platforma, tranzaktsii. // mining-cryptocurrency URL: <https://mining-cryptocurrency.ru/blockchain/> (data obrashcheniya: 26.06.18).
- [11] Andreas M. Antonopoulos. 7. The Blockchain // Mastering Bitcoin. — O'Reilly Media, Inc., 2014.
- [12] Pedro Franco. The Blockchain // Understanding Bitcoin: Cryptography, Engineering and Economics. — John Wiley & Sons, 2014.
- [13] Bitcoin: A Peer-to-Peer Electronic Cash System // Bitkoyн URL: <https://bitcoin.org/bitcoin.pdf> (data obrashcheniya: 22.06.2018).
- [14] MIT vypustil 100 diplomov na blokcheyne // bits.media URL: <https://bits.media/news/mit->

- vypustil-100-diplomov-na-blokcheyne/ (data obrashcheniya: 26.06.18).
- [15] 100 Diplomas: MIT Issues Graduate Certificates on a Blockchain App // coindesk.com URL: <https://www.coindesk.com/100-diplomas-mit-issues-graduate-certificates-on-a-blockchain-app/> (data obrashcheniya: 26.06.18).
- [16] C Sharp // wikipedia.org URL: https://ru.wikipedia.org/wiki/C_Sharp (data obrashcheniya: 26.06.18).
- [17] Vvedeniye v Windows Forms // metanit.com URL: <https://metanit.com/sharp/windowsforms/1.1.php> (data obrashcheniya: 26.06.18).
- [18] Rabota s formami // metanit.com URL: <https://metanit.com/sharp/windowsforms/2.1.php> (data obrashcheniya: 26.06.18).
- [19] Primer svyazannogo spiska // C++ dlya nachinayushchikh URL: http://mycpp.ru/cpp/book/c05_11.html (data obrashcheniya: 26.06.18).
- [20] Obshchiye svedeniya ob elemente upravleniya DataGridView (Windows Forms) // microsoft.com URL: <https://docs.microsoft.com/ru-ru/dotnet/framework/winforms/controls/datagridview-control-overview-windows-forms> (data obrashcheniya: 26.06.18).
- [21] Klass MessageBox // msdn.microsoft.com URL: [https://msdn.microsoft.com/ru-ru/library/system.windows.forms.messagebox\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/system.windows.forms.messagebox(v=vs.110).aspx) (data obrashcheniya: 26.06.18).
- [22] SHA2//<https://ru.wikipedia.org/wiki/SHA-2>(data obrashcheniya: 26.06.18).
- [23] SHA-256 – zhivaya klassika sredi algoritmov kheshirovaniya // tutdenegki.com URL: <http://tutdenegki.com/sha-256> (data obrashcheniya: 26.06.18).
- [24] Funktsii v Si-sharp. Operator return // mycsharp.ru URL: http://mycsharp.ru/post/13/2013_05_21_funkcii_v_si-sharp_operator_return.html (data obrashcheniya: 26.06.18).
- [25] Klassy // professorweb.ru URL: https://professorweb.ru/my/csharp/charp_theory/level_5/5_1.php (data obrashcheniya: 26.06.18).
- [26] while (Spravochnik po C#) // microsoft.com URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/while> (data obrashcheniya: 26.06.18).