

OBJECT-ORIENTED APPROACH IN COMPUTER MODELING OF THE ENCRYPTION ALGORITHM BASED ON NONPOSITIONAL POLYNOMIAL NOTATION SYSTEM

S. E. Nyssanbayeva, M. M. Magzom

Institute of Information and Computational Technologies
050010, Almaty, Republic of Kazakhstan

In this paper a computer implementation of the models of the nonconventional encryption algorithm, based on nonpositional polynomial notation system, is presented. An important stage of mathematical and computer modeling is a transformation of a mathematical model in a computer program. Development and testing of software for scientific research often takes a long time. One possible solution is the use of object-oriented programming. This approach enables code reuse while encapsulation of data and templates provide reliable code. The use of OOP is particularly advantageous in scientific research programs that include a parallel processing.

This paper considers a model of block cipher algorithm developed using nonpositional polynomial notation systems (NPNs) or a polynomial residual notation system (RNS). Classical modular arithmetic is based on the Chinese remainder theorem, which states that any number can be represented by their remainders (residues) from its division by the base numbers systems, which are formed pairwise prime numbers. In polynomial RNS moduli are represented by irreducible polynomials with coefficients over the $GF(2)$. The usage of NPNs allows improving durability and efficiency of nonpositional cryptographic algorithms without increasing the length of secret key.

Improved efficiency is provided by the rules of NPNs in which all arithmetic operations can be performed in parallel to the base module NPNs. In nonpositional cryptosystems the cryptostrength is characterized by a complete secret key. Cryptostrength in this case depends not only on the length of a key sequence, but also on choice of a system of polynomial bases. With the growth of the order of irreducible polynomials with binary coefficients, their number also grows rapidly. The greater the length of the input block, the more choices of working systems bases are possible. Therefore, the cryptostrength of the proposed encryption algorithm against bruteforce attack significantly increases with the length of the electronic message. During the development of the nonpositional encryption algorithm different designs of the Feistel scheme and encryption modes are investigated.

In this paper the methods of object-oriented programming, that simplify the research process of developed models, are described. The use of object-oriented approach and design patterns makes the program design more flexible. In particular, this makes it possible to easily change the classes that define the components of the model of the encryption algorithm. The biggest obstacle lies in hard coded information about which model configuration is used. With the creational patterns, there are different ways to get rid of the explicit reference to the specific code that implements the functions of a cryptosystem.

Basic design patterns used to create the foundations of the program are described. Application of Abstract Factory, Singleton, Strategy design patterns is shown. Furthermore the structure of the developed software is described. The functionality of the „polynomial“ and „crypto“ packaged is explained, and included classes BinaryPolynomial, BinaryPolynomialMath, BinaryHelpers, SimpleCipher, SimpleFeistel are shown.

The use of the Java platform during the computer implementation makes it possible to use the software implementation of the nonconventional encryption algorithm in a wide range of computing

devices and operating systems During software implementation of developed models, the statistical characteristics of the resulting ciphertexts were analyzed by using statistical test suit.

Carried out the analysis of a computer program that implements the functions of generating a complete encryption key and performs encryption using a block cipher modes.

The research of the possibility of implementing Feistel scheme and encryption modes helps to investigate the practical usability of the developed models. Computer modelling of the nonpositional encryption algorithm allows to develop recommendations for its application.

Key words: cryptographic system, encryption algorithm, modular arithmetic, computer modeling.

References

1. Laxmikant V.K., Sanjeev K. CHARM++. A Portable Concurrent Object Oriented System Based On C++ // OOPSLA 93, P. 91–108.
2. Biyashev R., Kalimoldayev M., Nyssanbayeva S., Magzom M. Development of an encryption algorithm based on nonpositional polynomial notations // WCNSSP2016, June 26–27, 2016. Chiang Mai, Thailand. P. 243–245.
3. Biyashev R., Nyssanbayeva S., Begimbayeva Ye., Magzom M. Building modified modular cryptographic systems // International Journal of Applied Mathematics and Informatics. 2015. Vol. 9. P. 103–109.
4. Schinianakis D., Stouraitis T. Residue Number Systems in Cryptography: Design, Challenges, Robustness // Secure System Design and Trustable Computing / Springer. 2016.
5. Introduction to Algorithms (sec. ed.) / T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. MIT Press and McGraw-Hill, 2001. P. 873–876.
6. Biyashev R. Razrabotka i issledovanie metodov skvoznoho povyshenija dostovernosti v sistemah obmena dannymi raspredelennyh ASU: doc. thesis, Moscow, 1985. P. 328. (In russian).
7. Biyashev R., Nyssanbayeva S. Algorithm for creation a digital signature with error detection and correction // Cybernetics and System Analysis. 2012. N 4. P. 14–23.
8. Kapalova N., Nyssanbayeva S., Khakimov R. Irreducible polynomials over the field $GF(2^n)$ // Proceedings of „KAKHAK“ scientific-technical society, Almaty, Kazakhstan, 2013, 1. P. 17–28.
9. Nyssanbayev S., Magzom M. Model' netradicionnogo algoritma shifrovaniya na osnove vlozhennyh setej Fejstelja // Vestnik KazNTU. 2016. N 4. (In russian).
10. Recommendation for Block Cipher Modes of Operation // NIST Special Publication 800-38A. 2001. P. 10.
11. Gamma E., Helm R., Johnson R., Vlissides J. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 2001. P. 395.
12. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications // NIST Special Publication 800-22. 2001. P. 154.

ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ ПОДХОД ПРИ КОМПЬЮТЕРНОМ МОДЕЛИРОВАНИИ АЛГОРИТМА ШИФРОВАНИЯ НА БАЗЕ НЕПОЗИЦИОННОЙ ПОЛИНОМИАЛЬНОЙ СИСТЕМЫ СЧИСЛЕНИЯ

С. Е. Нысанбаева, М. М. Магзом

Институт информационных и вычислительных технологий КН МОН РК
050010, Алма-Ата, Республика Казахстан

УДК 004.056.5

В работе рассматривается компьютерная реализация моделей нетрадиционного алгоритма шифрования, основанного на непозиционной полиномиальной системе счисления. Описаны методы объектно-ориентированного программирования, упрощающие процесс исследования разработанных моделей. Проведен анализ компьютерной программы, реализующей функции генерации полного ключа шифрования и выполняющей шифрование с использованием режимов блочных шифров.

Ключевые слова: криптографическая система, алгоритм шифрования, модулярная арифметика, компьютерное моделирование.

Введение. Важным этапом математического и компьютерного моделирования является преобразование математической модели в готовую компьютерную программу. Разработка и отладка программного обеспечения (ПО) для проведения научных исследований часто занимает много времени. Для уменьшения усилий, затрачиваемых на решение задач программирования, возможно использовать современные методы разработки ПО, которые делают реализацию приложения более понятной и эффективной.

Одним из возможных решений является использование объектно-ориентированного программирования (ООП). Данный подход предоставляет возможность повторного использования кода, а инкапсуляция данных и шаблоны предоставляют надежный код. Использование ООП является особенно предпочтительным в научных программах, включающих параллельную обработку [1].

Рассматриваемые в данной работе модели алгоритма блочного шифрования разработаны с использованием непозиционной полиномиальной системы счисления (НПСС) или полиномиальной системы остаточных классов (СОК) [2]. В работе [3] описаны модификации данных моделей нетрадиционного алгоритма шифрования с использованием сети Фейстеля и режимов блочного шифра.

В классической системе остаточных классов каждое число, многоразрядное в позиционной системе счисления, представляется в виде нескольких малоразрядных позиционных чисел, которые являются остатками, полученными от деления исходного числа на взаимно простые основания [4]. В обычной позиционной двоичной системе при выполнении арифметических операций образуются переносы в следующий старший разряд, вследствие чего

Работа выполнена при поддержке программно-целевого финансирования научно-технических программ и проектов Комитета науки МОН РК № 0128/ПЦФ.

необходимо выполнять данные операции последовательно по разрядам. В СОК существует возможность распараллелить этот процесс. В соответствии с правилами, все операции над остатками по каждому основанию можно выполнять отдельно и независимо, т. е. параллельно. С учетом того, что операции проводятся над данными меньшей разрядности, чем входной блок, то скорость обработки также увеличивается.

В данной работе описывается разработка компьютерной модели алгоритма симметричного шифрования, основанного на базе полиномиальной системы остаточных классов.

1. Алгоритм симметричного шифрования на базе НПСС. Аналогично китайской теореме об остатках [5], в полиномиальных системах счисления в остаточных классах любой полином может быть представлен своими остатками (вычетами) от деления на систему оснований, состоящей из неприводимых многочленов над полем $GF(2)$ [6, 7].

Для формирования НПСС при шифровании блока длиной N бит из множества всех неприводимых многочленов степени не выше значения N выбираются рабочие основания

$$p_1(x), p_2(x), \dots, p_S(x). \quad (1)$$

Все выбираемые основания должны отличаться друг от друга, даже если они являются неприводимыми полиномами одной степени. Тогда в этой системе любой многочлен степени меньше суммы степеней всех рабочих оснований (1) имеет единственное представление в виде последовательности остатков (вычетов) от деления его на данные основания. Таким образом, блок открытого текста и ключевая последовательность длиной N бит могут быть представлены в виде последовательностей вычетов $F(x) = (\alpha_1(x), \alpha_2(x), \dots, \alpha_S(x))$ и $G(x) = (\beta_1(x), \beta_2(x), \dots, \beta_S(x))$ соответственно, полученных в результате деления по рабочим основаниям системы. Покажем пример простого преобразования открытого текста в шифр.

Шифрованное сообщение формируется в результате умножения многочленов $F(x)$ и $G(x)$:

$$F(x)G(x) \equiv H(x)(\text{mod} P(x)), \quad (2)$$

т. е. может быть представлено в виде остатков от деления произведений $\alpha_i(x)\beta_i(x)$ на соответствующие основания $p_i(x)$.

Для расшифрования необходимо вычислить обратный (инверсный) многочлен $G^{-1}(x) = (\beta_1^{-1}(x), \beta_2^{-1}(x), \dots, \beta_S^{-1}(x))$. Тогда исходное сообщение восстанавливается по сравнению:

$$F(x) \equiv G^{-1}(x)H(x)(\text{mod} P(x)). \quad (3)$$

Секретный ключ, используемый для шифрования, называется полным. Полный ключ состоит из ключевой псевдослучайной последовательности и выбранной системы полиномиальных оснований, которая также держится в секрете. С ростом порядка неприводимых многочленов с двоичными коэффициентами их количество стремительно растет, в связи с этим очевиден широкий выбор полиномиальных оснований. Количество вариантов выбора НПСС существенно возрастает с увеличением длины шифруемого блока [8]. В связи с этим, в алгоритме шифрования, основанном на НПСС, секретные параметры шифра зависят не только от длины ключевой последовательности, но и от выбранной системы полиномиальных оснований, а также от порядка расположения оснований в системе.

В работе [3] были описаны модификации нетрадиционного алгоритма шифрования с использованием сети Фейстеля в качестве пред- и постобработки блока шифруемых данных. Применение указанных выше свойств алгоритма шифрования при генерации раундовых

ключей приводит к неравномерному изменению внутренних свойств сети Фейстеля, что усложняет анализ свойств моделей шифра.

В отличие от традиционной сети Фейстеля, где входными данными является открытый текст сообщения, в модели с постобработкой на вход подается битовая последовательность шифротекста, получаемая при шифровании нетрадиционным алгоритмом на базе НПСС.

В модели с предобработкой блок открытого текста предварительно шифруется по классической схеме Фейстеля, после чего преобразуется нетрадиционным методом шифрования.

Кроме этого, была разработана модель алгоритма шифрования, которая повторяет структуру классической сети Фейстеля, но в которой раундовая F -функция преобразует подблок входных данных с помощью нетрадиционного метода шифрования. Функция шифрования F подблока может зависеть не только от раундового ключа $K(i)$, но и от выбранной системы оснований. В этом случае данная функция будет называться гетерогенной. Применение гетерогенных сетей может значительно улучшить характеристики шифра, поскольку неравномерное изменение внутренних свойств сети в пределах допустимых границ делает изучение свойств шифра достаточно затруднительным занятием.

В [9] показана структура модели алгоритма шифрования с использованием вложенной сети Фейстеля. В данной модели функция преобразования F также представляет собой сеть Фейстеля.

Во всех моделях для улучшения статистических свойств получаемых криптограмм используется режим шифрования. Режимы шифрования используются для модификации процесса шифрования так, чтобы результат шифрования каждого блока был уникальным вне зависимости от шифруемых данных и не позволял сделать какие-либо выводы об их структуре. Это обусловлено, прежде всего, тем, что блочные шифры шифруют данные блоками фиксированного размера, и поэтому существует потенциальная возможность утечки информации о повторяющихся частях данных шифруемых на одном и том же ключе.

В данной модели применяется режим Cipher Block Chaining [10] — режим сцепления блоков шифра. Преобразование выполняется следующим образом: каждый блок открытого текста складывается по модулю 2 с результатом шифрования предыдущего блока. Таким образом, результаты шифрования предыдущих блоков влияют на шифрование следующих блоков.

При этом в начале шифрования используется вектор инициализации для того, чтобы любое сообщение было уникальным. В связи с этим вектор инициализации должен быть случайным числом. Его не обязательно хранить в секрете, можно передавать его вместе с сообщением.

2. Программная реализация моделей с использованием объектно-ориентированного подхода. При компьютерном моделировании алгоритма шифрования необходимо предусматривать возможности изменения программы и поставленных задач.

Для решения задачи в процедурном программировании необходимо создать различные функции, реализующие функции генерации полного ключа шифрования, и выполняющие шифрование с использованием режимов блочных шифров для каждой из необходимых конфигураций моделей шифра. В функциях должны быть жестко определены параметры данных моделей. Для того чтобы изменить структуру, придется изменить саму функцию, либо заменив ее (то есть полностью переписав заново), либо непосредственно модифици-

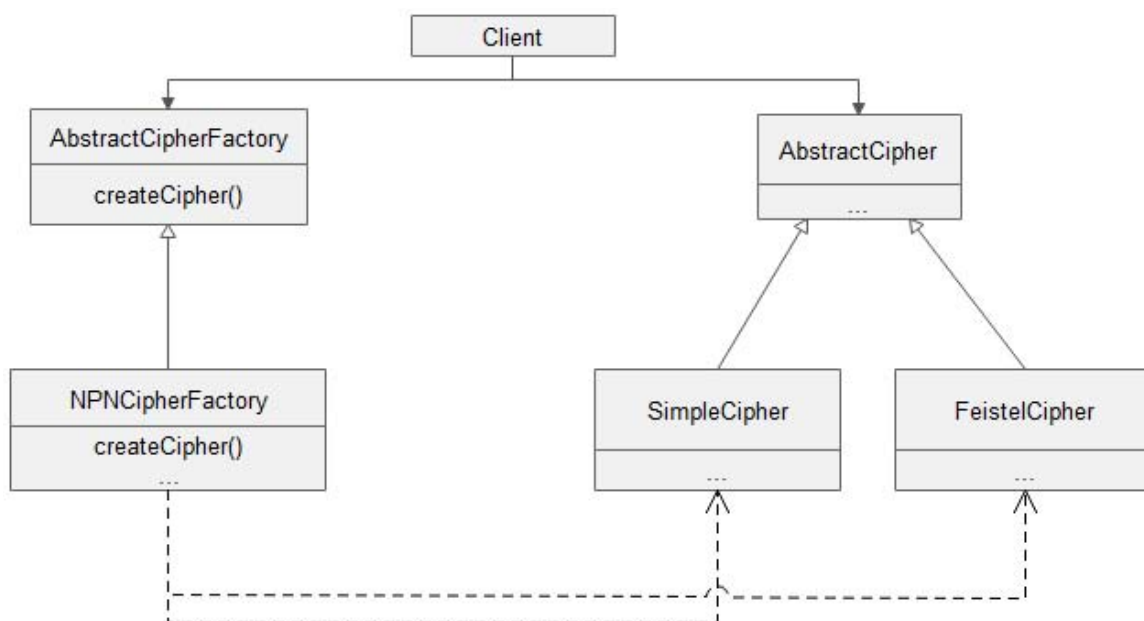


Рис. 1. Структура фабрики моделей алгоритма шифрования

ровав ее фрагменты. Оба пути чреваты ошибками и не способствуют повторному использованию.

Использование объектно-ориентированного подхода и шаблонов проектирования позволяют сделать дизайн более гибким, хотя и необязательно меньшим по размеру. В частности, это дает возможность легко изменять классы, определяющие компоненты моделей алгоритма шифрования.

Самое серьезное препятствие лежит в жестко указанной в коде информации о том, какая конфигурация модели реализуется. С помощью порождающих шаблонов можно различными способами избавиться от явных ссылок на конкретные функции кода [11], реализующие функционал моделей криптосистемы.

Рассмотрим основные шаблоны проектирования, использованные при создании основы программы.

Шаблон проектирования „Factory“ („Фабрика“) инкапсулирует создание одного из нескольких связанных классов. Назначение данного шаблона проектирования — предоставлять интерфейс для создания семейств взаимосвязанных или взаимозависимых объектов, не специфицируя их конкретных классов.

Чтобы в приложении можно было использовать различные конфигурации, в нем не должны быть жестко закодированы параметры алгоритма.

Если реализация и инстанцирование классов для конкретной конфигурации разбросано по всему приложению, то изменить методы расчета впоследствии будет нелегко.

Для решения этой проблемы определен абстрактный класс `AbstractCipherFactory`, в котором объявлен интерфейс для создания всех разработанных моделей алгоритма шифрования. Клиенты используют этот интерфейс для получения экземпляров моделей шифра, но при этом ничего не знают о том, какие именно классы используют. Стало быть, клиенты остаются независимыми от выбранной конфигурации. На рис. 1 показана структура данного шаблона.

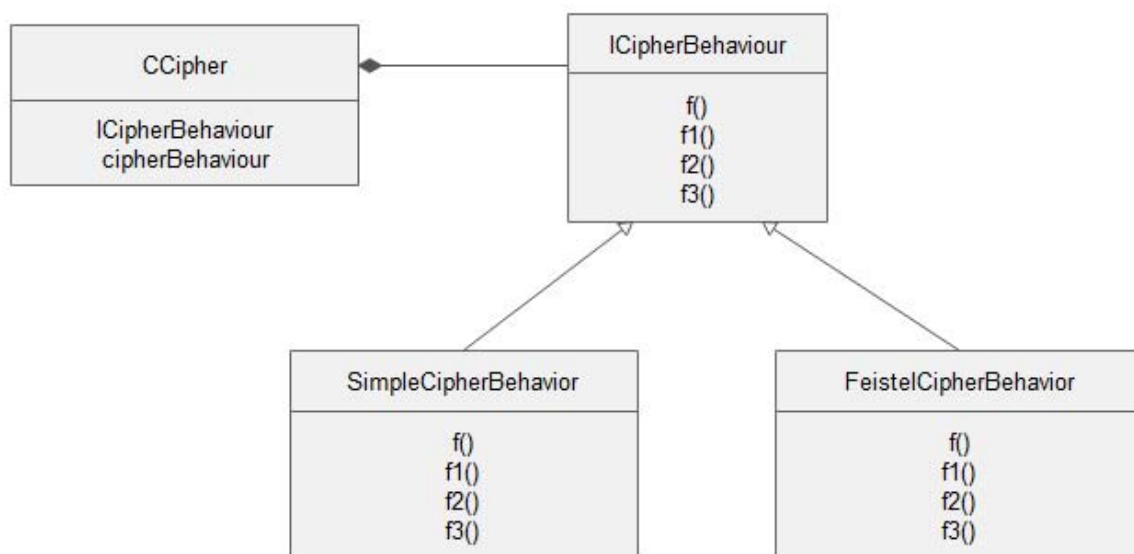


Рис. 2. Структура шаблона „Поведение“

Как правило, приложению нужен только один экземпляр класса ConcreteFactory на каждое семейство продуктов. Необходимо обеспечить конструирование экземпляра один раз, а затем обеспечить обращение только к нему.

Эта задача решается с помощью шаблона „Singleton“ („Одиночка“), который гарантирует, что у класса есть только один экземпляр, и предоставляет к нему глобальную точку доступа. Данный шаблон проектирования устроен так, что тот единственный экземпляр, который имеется у класса, — самый обычный, но больше одного экземпляра создать не удастся. Чаще всего для этого ограничивают доступ к операции, создающей экземпляры, пряча ее за операцией класса (т. е. за статической функцией-членом или методом класса), которая гарантирует создание не более одного экземпляра. Данная операция имеет доступ к переменной, где хранится уникальный экземпляр, и гарантирует инициализацию переменной этим экземпляром перед возвратом ее клиенту. При таком подходе можно не сомневаться, что экземпляр „Одиночки“ будет создан и инициализирован перед первым использованием.

Клиенты осуществляют доступ к „Одиночке“ исключительно через функцию-член getInstance. Переменная instance инициализируется нулем, а статическая функция-член getInstance возвращает ее значение, инициализируя ее уникальным экземпляром, если в текущий момент оно не определено. Функция getInstance использует отложенную инициализацию: возвращаемое ей значение не создается и не хранится вплоть до момента первого обращения.

Шаблон проектирования „Strategy“ („Поведение“) предоставляет семейство алгоритмов, инкапсулирует каждый из них и делает их взаимозаменяемыми. „Поведение“ позволяет изменять алгоритмы независимо от клиентов, которые ими пользуются. Этот шаблон позволяет менять поведение объекта, не меняя его структуру. Таким образом можно изменять типы модели алгоритма шифрования — соответственно и поведение шифра, на этапе выполнения программы без необходимости повторной компиляции и сборки исходного текста. Схема данного шаблона показана рис. 2.

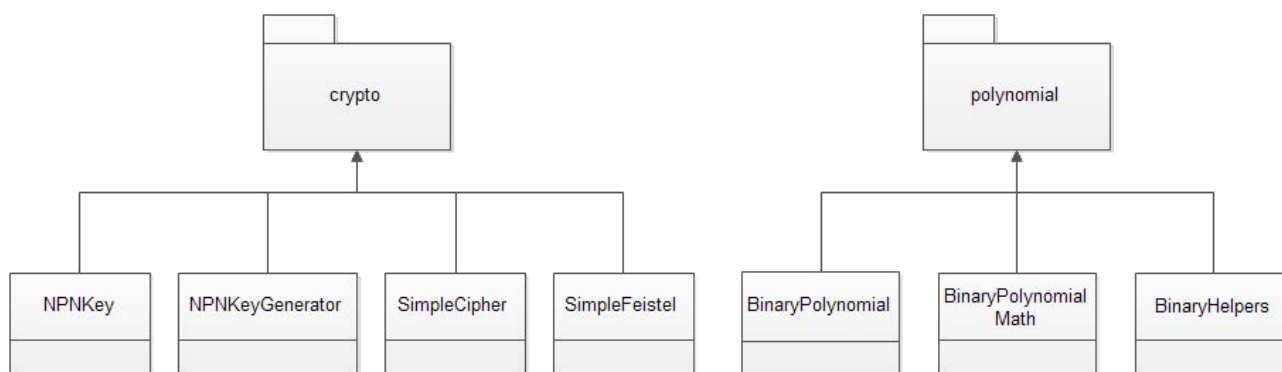


Рис. 3. Структура компьютерной реализации разработанных моделей

Разрабатываемые модели были реализованы с использованием языка программирования Java.

На рис. 3 показана диаграмма с структурой разработанной программы.

В пакете `polynomial` созданы классы для работы с многочленами над полем $GF(2)$. Был разработан класс `BinaryPolynomialMath`, реализующий арифметические операции с бинарными полиномами. Данный класс является основой для компьютерной реализации разработанных моделей.

Разработан класс `BinaryPolynomial`, описывающий многочлен над полем $GF(2)$. Методы данного класса позволяют создавать экземпляры многочленов заданного порядка, переводить форму отображения многочлена из двоичного в текстовый, определять неприводимость данного полинома.

Класс `BinaryHelpers` реализует вспомогательные методы для работы с двоичными данными.

Разработанные модели алгоритмов шифрования реализованы в пакете `crypto`. Также туда входят классы для генерации полного ключа шифрования, использующие различные источники энтропии операционной системы.

Классы `SimpleCipher` и `SimpleFeistel` реализуют алгоритм шифрования на базе НПСС с применением сети Фейстеля и режимов блочного шифра. Программа позволяет задавать параметры для управления размером блока, количеством раундов и использования режима шифрования.

Анализ статистических характеристик получаемых шифртекстов проводится путем использования набора статистическим тестов [12].

Заключение. Цель проводимых исследований заключается в анализе возможностей практического применения алгоритма шифрования, разработанного на базе непозиционных полиномиальных систем счисления и исследовании эффективности использования классических структур и режимов блочных шифров. В связи с этим разработано несколько моделей алгоритма шифрования и их компьютерной реализации.

Было проведено исследование по применению шаблонов проектирования при создании компьютерных моделей. Разработаны рекомендации по использованию конкретных шаблонов проектирования и проведены эксперименты по проверке эффективности применения объектно-ориентированного подхода при моделировании нетрадиционного алгоритма шифрования.

Использование платформы Java при компьютерной реализации дает возможность использовать программную реализацию нетрадиционного алгоритма шифрования в широком спектре вычислительных устройств и операционных систем. Реализация библиотеки криптоалгоритма позволит в дальнейшем внедрять данный алгоритм шифрования в различные клиент-серверные системы, веб-приложения и мобильные устройства.

Список литературы

1. Laxmikant V.K., Sanjeev K. CHARM++. A Portable Concurrent Object Oriented System Based On C++ // OOPSLA 93, P. 91–108.
2. Biyashev R., Kalimoldayev M., Nyssanbayeva S., Magzom M. Development of an encryption algorithm based on nonpositional polynomial notations // WCNSP2016, June 26–27, 2016. Chiang Mai, Thailand. P. 243–245.
3. Biyashev R., Nyssanbayeva S., Begimbayeva Ye., Magzom M. Building modified modular cryptographic systems // International Journal of Applied Mathematics and Informatics. 2015. Vol. 9. P. 103–109.
4. Schinianakis D., Stouraitis T. Residue Number Systems in Cryptography: Design, Challenges, Robustness // Secure System Design and Trustable Computing / Springer. 2016.
5. Introduction to Algorithms (sec. ed.) / Т. Н. Cormen, С. Е. Leiserson, R. L. Rivest, С. Stein. MIT Press and McGraw-Hill, 2001. P. 873–876.
6. Бияшев Р. Разработка и исследование методов сквозного повышения достоверности в системах обмена данными распределенных АСУ: дис. докт. тех. наук. Москва, 1985.
7. Бияшев Р., Нысанбаева С. Алгоритм формирования электронной цифровой подписи с возможностью обнаружения и исправления ошибки // Кибернетика и системный анализ. 2012. № 4. С. 14–23.
8. Капалова Н., Нысанбаева С., Хакимов Р. Неприводимые полиномы над полем $GF(2^n)$ // Известия научно-технического общества „КАХАК“. 2013. № 1. С. 17–28.
9. Нысанбаева С., Магзом М. Модель нетрадиционного алгоритма шифрования на основе вложенных сетей Фейстеля // Вестник КазНТУ. 2016. № 4.
10. Recommendation for Block Cipher Modes of Operation // NIST Special Publication 800-38A. 2001. P. 10.
11. Приемы объектно-ориентированного проектирования. Паттерны проектирования // Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Н.СПб: Питер, 2001.
12. A statistical test suite for random and pseudorandom number generators for cryptographic applications // NIST Special Publication 800-22. 2001. P. 154.



Нысанбаева Сауле Еркебулановна — д-р техн. наук, ассоциированный профессор, гл. науч. сотр. Института информационных и вычислительных технологий КН МОН РК; тел.: +77017743730.

Сауле Нысанбаева окончила механико-математический факультет Казахского государственного университета им. С.М. Кирова (КазГУ, г. Алма-Ата) в 1971 г. В 1986 г. за-

щитила кандидатскую диссертацию по специальности 01.04.14 „Математическое моделирование тепловых режимов кристаллизации переохлажденных жидкостей“ на соискание ученой степени кандидата физико-математических наук. В 2009 г. защитила диссертацию „Разработка и исследование криптографических систем на базе непозиционных полиномиальных систем“ на соискание ученой степени доктора технических наук. С октября 1971 г. года работала в КазГУ на кафедре прикладной матема-

тики и в проблемной лаборатории математического моделирования. С февраля 1994 г. по август 2001 г. работала в Институте проблем информатики и управления (ныне Институт информационных и вычислительных технологий) Министерства образования и науки Республики Казахстан ученым секретарем. С декабря 2003 г. — сотрудник лаборатории информационной безопасности ИПИУ. Тематика научных исследований — разработка и исследование алгоритмов и средств криптографической защиты информации с использованием модулярной арифметики, хранимой и распространяемой в инфо-коммуникационных системах.

Saule Nyssanbayeva graduated from the Mechanics and Mathematics Faculty of the Kazakh State University named after S. M. Kirov in 1971. In 1986, she received her candidate degree on speciality 01.04.14 „Mathematical modeling of the thermal modes of crystallization of supercooled liquids“. In 2009 she defended her thesis „Research and development of cryptographic systems based on nonpositional polynomial systems“ and received a degree of Doctor of Technical Sciences. Since October 1971, she worked in the Kazakh State University in the chair of applied mathematics and in the laboratory of mathematical modeling. From February 1994 to August 2001, she worked as a scientific secretary at the Institute of Problems of Informatics and Control (now the Institute of Informational and Computational Technologies) of Ministry of Education and Science of the Republic of Kazakhstan. Since December 2003 — researcher in the information security laboratory. Subject of research — development and analysis of

algorithms and means of cryptographic protection of information using modular arithmetic.



Магзом Мирас Мухтарулы — мл. науч, сотр. Института информационных и вычислительных технологий КН МОН РК; e-mail: magzomxzn@gmail.com.

Мирас Магзом получил степень магистра технических наук по специальности „Вычислительная техника и программное обеспечение“ в 2014 г. в Алматинском университете энергетики и связи. С 2014 г. проходит обучение по совместной программе PhD в Институте информационных и вычислительных технологий МОН РК и Казахском национальном университете им. Аль-Фараби по специальности „Вычислительная техника и программное обеспечение“. С 2015 г. работает в лаборатории информационной безопасности Института информационных и вычислительных технологий.

Miras Magzom received his M. S. degree in Computer Science in 2014 from Almaty University of Power Engineering and Telecommunications. Since 2014 studies PhD in Computer Science by the joint program in Institute of Informational and Computational Technologies of Ministry of Education and Science of the Republic of Kazakhstan and Al-Farabi Kazakh National University. Since 2015 works in the Laboratory of Informational Security in Institute of Informational and Computational Technologies.

Дата поступления — 26.08.2016