УДК 511.17

И.В. Лысенко, В.В. Бородавка

Национальный аэрокосмический университет имени Н.Е. Жуковского «ХАИ», Харьков

РАЗРАБОТКА ТЕОРЕТИКО-ЧИСЛОВОГО ТУЛБОКСА ДЛЯ СИСТЕМЫ КОМПЬЮТЕРНОЙ МАТЕМАТИКИ MATLAB

Представлены результаты сравнительного анализа наиболее популярных систем компьютерной математики (Matlab, Mathematica, Maple, Mathcad) на предмет возможности решения в их рамках теоретико-числовых задач и вычисления функций элементарной теории чисел. Описаны возможности разработанного теоретико-числового тулбокса Number Theory Toolbox для системы Matlab , представленного набором функций для выполнения теоретико-числовых преобразований, относящихся к прикладной криптологии.

Ключевые слова: *Matlab, функции теории чисел, теоретико-числовой тулбокс.*

Введение

В практике инженерной и исследовательской деятельности многие прикладные задачи, допускающие формализацию, могут быть решены с помощью систем компьютерной математики (СКМ) или, как иногда говорят, систем компьютерной алгебры, представляющих собой специализированное программное обеспечение для выполнения математических расчётов самой разной направленности. К числу наиболее популярных СКМ относятся Matlab, Mathematica, Maple, Mathcad [1-2]. Так, например, в [3] сделан обзор возможностей упомянутых СКМ с точки зрения решения задач оптимизации.

Следует, очевидно, ожидать от СКМ возможности вычисления в их рамках базовых функций элементарной теории чисел и решения теоретико-числовых задач, относящихся к прикладной криптологии. Проведенный анализ показывает, что наибольшими возможностями в этом отношении обладают СКМ Mathematica и Maple.

В этой связи **цель статьи** состоит в том, чтобы, во-первых, представить результаты анализа СКМ Matlab, Mathematica, Maple, Mathcad на предмет возможности решения в их рамках теоретико-числовых задач и вычисления функций элементарной теории чисел, относящихся к прикладной криптологии, а, во-вторых, описать возможности разработанного теоретико-числового тулбокса для СКМ Matlab (Number Theory Toolbox).

Сравнительная характеристика возможностей СКМ по решению задач элементарной теории чисел

В СКМ Mathcad, Matlab, Mathematica, Maple присутствуют следующие встроенные теоретико-числовые функции, общие для них: *primes*, *isprime*, *factor*, *gcd*, *lcm*, *mod*. Первые три из них являются функциями одного аргумента - натурального числа n и связаны с простыми числами. Так, функция *primes* возвращает строку простых чисел, меньших или равных n, а вторая позволяет установить, является ли данное число простым. Функция *factor* решает задачу факторизации – возвращает строку, содержащую простые множители числа n. Функции *gcd* и *lcm* находят наибольший общий делитель (НОД) и наименьшее общее кратное (НОК) двух чисел соответственно и, наконец, функция *mod*(x,y) возвращает остаток от деления x на y. Что касается простых чисел, в СКМ Mathematica и Maple, в отличие от Mathcad и Matlab содержатся встроенные функции, позволяющие: находить простое число с заданным порядковым номером, наименьшее простое число, превышающее заданное число n и наибольшее простое число, не превышающее его, а также все простые делители числа n. Кроме того, встроенные функции СКМ Mathematica позволяют:

- находить следующее простое число, которое расположено после n (функция *NextPrime*);
- находить совершенное число по его порядковому номеру (функция *PerfectNumber*);
- находить символ Кронекера (функция *KroneckerSymbol*);
- возводить число в степень по модулю (функция *PowerMod*);
- находить функцию Эйлера числа (функция *EulerPhi*);
- находить функцию Мебиуса числа (функция *MoebiusMu*);
- находить символ Якоби и символ Лежандра числа (функция *JacobiSymbol*);
- находить порядковый номер числа по модую (функция *MultiplicativeOrder*);
- выбрать случайное простое число (функция *RandomPrime*).

В состав СКМ Mathematica входит пакет Number Theory, который содержит все выше перечисленные функции и насчитывает всего 56 функций для теоретико-числовых вычислений.

В состав СКМ Maple также входит пакет Number Theory, который содержит команды, используемые для исследования свойств натуральных и целых

чисел. Всего данный пакет содержит 42 функции для теоретико-числовых вычислений. Функции пакета Number Theory в СКМ Maple позволяют:

- проверить, является ли данное число числом Мерсенна (функция *IsMersenne*);
- найти количество простых чисел меньше заданного числа (функция *PrimeCounting*);
- находить квадратичные вычеты числа (функция *QuadraticResidue*);
- вычислять количество простых сомножителей целого числа n с учетом кратности (функция *NumberOfPrimeFactors*);
- находить сумму делителей числа (функция *SumOfDivisors*);
- находить первообразный корень по модулю (функция *PrimitiveRoot*);
- решать уравнения или неравенства Туэ (функция *ThueSolve*);
- решать задачи о сумме двух квадратов (функция *SumOfSquares*).

сравнительная характеристика СКМ с точки зрения наличия в них теоретико-числовых функций, имеющих применение в криптографии, представлены в табл. 1 – 4 [4 – 9].

Таблица 1

Сравнительная таблица возможностей СКМ

Системы компьютерной математики	Задачи элементарной теории чисел				
	ФЭ	ФР1	ФН1	ФН2	ФН2
Maple	+	—	+	—	+
Matlab	—	—	—	—	—
Mathcad	—	—	—	—	—
Mathematica	+	—	+	+	+

Таблица 2

Сравнительная таблица возможностей СКМ

Системы компьютерной математики	Задачи элементарной теории чисел				
	ФМ	ФЛ	ФЯ	ФСМ	ФПМ
Maple	+	+	+	—	—
Matlab	—	—	—	—	—
Mathcad	—	—	—	—	—
Mathematica	+	—	+	+	—

Таблица 3

Сравнительная таблица возможностей СКМ

Системы компьютерной математики	Задачи элементарной теории чисел				
	ФОМ	ФОБ	ФР2	ФОП	ФНК
Maple	+	—	—	+	—
Matlab	—	—	—	—	—
Mathcad	—	—	—	—	—
Mathematica	—	—	—	+	+

Таблица 4

Сравнительная таблица возможностей СКМ

Системы компьютерной математики	Задачи элементарной теории чисел			
	ФРК	ФПП	ФНОД	ФНОК
Maple	+	—	—	—
Matlab	—	—	—	—
Mathcad	—	—	—	—
Mathematica	+	+	+	+

В табл. 1 – 4 обозначены функции: Эйлера (ФЭ), решения сравнений первой степени (ФР1), нахождения делителей числа (ФН1), их количества (ФН2) и суммы (ФН3); Мебиуса (ФМ); нахождения символов Лежандра (ФЛ) и Якоби (ФЯ); возведения в степень по модулю (ФСМ) и поиска квадратичных вычетов по модулю (ФПМ); определения – является ли число квадратичным вычетом по модулю (ФОМ) или является ли простое число числом Блюма (ФОБ); решения квадратичного сравнения для случая, когда простое число является числом Блюма (ФР2); определения показателя числа по модулю (ФОП); нахождения обратного элемента в кольце по модулю (ФНК); решения системы сравнений первой степени на основе китайской теоремы об остатках (ФРК); определения простого числа с порядковым номером (ФПП), НОД для произвольного числа аргументов (ФНОД) и НОК для произвольного числа аргументов (ФНОК).

Теоретико-числовой тулбокс для СКМ Matlab (Number Theory Toolbox)

На данный момент в СКМ Matlab отсутствует тулбокс для теоретико-числовых вычислений. Разработанный тулбокс может быть внедрен как отдельный Toolbox в СКМ Matlab и использован для целей криптографии (как вспомогательных и составляющих функций для практической реализации, и обоснования стойкости криптографических средств), а также в учебных целях, в частности, при изучении курсов, связанных с математическими основами криптологии.

Разработанный тулбокс содержит ряд функций:

1. Функция *eulerfunc(n)* производит вычисление функции Эйлера для заданного натурального числа n – количества положительных целых чисел, меньших n, взаимно-простых с n. В ходе *тестирования* было установлено, что 16 – это максимальное количество разрядов для входного значения n. Данное ограничение связано с возможностями встроенной функции факторизации *factor(n)* обрабатывать данные разрядности, не больше 16.

2. Функция *solv1powercongr(a,b,m)* предназначена для отыскания решения сравнений первой степени вида ax = b(mod m) с проверкой условия существования решений, а также определения их количества. При тестировании было установлено, что 15 – это максимальное количество разрядов для входных значений параметров a, b, m. При разрядности входных даннях, больших 16 десятичных разрядов, функция будет отображать количество решений сравнения, но без отображения результатов. Данное ограничение связано со встроенной функцией *dec2bin(n).*

3. Функция *divisors(n)* предназначена для нахождения делителей числа n и выводит список целых чисел, которые делят n (включая 1 и n). При тестировании функции было установлено, что 16 – это максимальное количество разрядов для входного значения n. Данное ограничение связано с упомянутым выше ограничением встроенной функции факторизации *factor(n)*.

4. Функция *divisorsnum(n)* предназначена для нахождения количества делителей числа n и выводит количество целых чисел, которые делят n (включая 1 и n). Входным параметром данной функции является целое число n. Тестирование аналогично предыдущей функции.

5. Функция *divisorssum(n)* предназначена для нахождения суммы делителей числа n и выводит сумму целых чисел, которые делят n (включая 1 и n). Входным параметром данной функции является целое число n. Тестирование аналогично предыдущей функции.

6. Функция *mobiusfunc(n)* производит вычисление функции Мебиуса, которая принимает значения: -1, 0, 1, а именно:

- $\mu(n) = 1$, если n не делится на квадрат простого числа и разложение n на простые множители состоит из чётного числа сомножителей;
- $\mu(n) = -1$, если n не делится на квадрат простого числа и разложение n на простые множители состоит из нечётного числа сомножителей;
- $\mu(n) = 0$, если n делится на квадрат простого числа.

При тестировании было установлено, что 12 – это максимальное количество разрядов для входного значения n. Данное ограничение связано с переполнением памяти при вычислениях, а также ограничениями встроенной функции *hist(n)*.

7. Функция *legendresymb(a,p)* производит вычисление символа Лежандра (a/p), где p – простое число, который принимает значения: -1, 0, 1, а именно:

- (a/p) = 0, если a делится на p;
- (a/p) = 1, если a является квадратичным вычетом по модулю p, т.е. существует такое целое x, что $x^2 = a \pmod p$;
- (a/p) = -1, если a является квадратичным невычетом по модулю p.

При тестировании было установлено, что единственным ограничением данной функции является сложность нахождения большого простого числа, а в некоторых случаях ограниченность встроенной функции *isprime(n)*.

8. Функция *jacobisymb* (a, n) производит вычисление символа Якоби (a, n). Символ Якоби является обобщением символа Лежандра, когда n – не обязательно простое число. При тестировании было установлено, что 16 – это максимальное количество разрядов для входных значений. Данное ограничение связано с ограничением на разрядность аргументов функции mod (a, b).

9. Функция *powermod (a,b,m)* производит вычисление значения возведения в степень о модулю $a^b \pmod m$. При тестировании функции было установлено, что 305, 305, 154 – это максимальное количество разрядов для входных значений a, b, m, соответственно.

10. Функция *quadraticresidues (n)* выводит квадратичные вычеты для заданного модуля. При тестировании было установлено, что 7 – это максимальное количество разрядов для входного значения n. Данное ограничение связано с переполнением памяти при вычислениях, а также ограничениями встроенной функции *unique(n)*.

11. Функция *isquadresidue(a,n)* позволяет ответить на вопрос, ли число *a* квадратичным вычетом по модулю n или нет. При тестировании было установлено, что 53 – это максимальное количество разрядов для аргументов функции. При разрядности входных данных больше 53, результат работы функции может быть некорректным.

12. Функция *fisblumprime(p)* позволяет установить, является ли число простым числом Блюма. При тестировании функции было установлено, что единственным ограничением является сложность нахождения большого простого числа, а в некоторых случаях – ограничения на разрядность аргумента встроенной функции *isprime(n)*.

13. Функция *solvquadblum(a,p)* предназначена для отыскания решения квадратичного сравнения для случая, когда простое число является числом Блюма. В этом случае, если $p = 3 \pmod 4$ – простое число и *a* – квадратичный вычет по модулю p, то сравнение $x^2 = a \pmod p$ имеет два решения: $x = \pm a^{(p+1)/4} \pmod p$. При тестировании было установлено, что ограничениями данной функции являются: сложность нахождения большого простого числа, а также – квадратичного вычета для него.

14. Функция *multorder(a,m)* предназначена для определения показателя числа *a* по модулю m, т.е. такого наименьшего числа r, для которого справедливо сравнение $a^r = 1 \pmod m$. При тестировании было установлено, что ограничениями данной функции являются оператор *while* и встроенная функция *mod(a,b),* у которых при разрядности входных данных, большем 12, вычисления и перебор занимают бесконечное количество времени.

15. Функция *modmultinv(a,m)* производит вычисление элемента, обратного элементу *a* в кольце вычетов по модулю m. При тестировании было установлено, что ограничением данной функции на разрядность входных данных является соответствующее ограничение функции *factor(n),* у которой максимально допустимая размерность входных данных равна 16 разрядов для входного значения n. В свою очередь, первый аргумент данной функции может иметь разрядность, равную 308, при максимальной разрядности второго аргумента, равной 15.

16. Функция *chineseremainder(r,a)* предназначена для отыскания решения системы сравнений с помощью китайской теоремы об остатках. Входными

параметрами данной функции являются два операнда r и n, где r – массив остатков, a – массив модулей линейных сравнений вида $x = r_i(\text{mod } n_i)$. При тестировании было установлено, что ограничением данной функции является встроенная функция *factor(n)* (эту встроенную функцию использует разработанная функция *eulerfunc(n),* которую в свою очередь использует разработанная функция *powermod(a,b,m)*, а разработанная функция *modmultinv(a,m)* использует *powermod(a,b,m)* для своих вычислений, поэтому автоматически ограничения встроенной функции *factor(n)* переходят и ко всем разработанным функциям), и что 13 – это максимальное количество разрядов для входных значений r и n.

17. Функция *prime(n)* предназначена для отыскания простого числа по его порядковому номеру n. При тестировании было установлено, что ограничениями данной функции является проблема выхода массива за допустимые пределы, а также ограничения встроенной функции *primes(n).*

18. Функция *igcd(a)* предназначена для определения НОД для произвольного числа аргументов a_1, a_2, ... , a_n, которые являются входными параметрами массива *a*. При тестировании было установлено, что ограничением данной функции является встроенная функция *gcd(a,b)*, которая при разрядности входных данных, большей 16, может давать неточный результат.

19. Функция *ilcm(a)* предназначена для определения НОК для произвольного числа аргументов a_1, a_2, ..., a_n, которые являются входными параметрами массива *a*. При тестировании было установлено, что ограничением данной функции является встроенная функция *lcm(a,b)*, которая при разрядности входных данных, большей 8, может давать неточный результат.

Каждая из вышеперечисленных функций была протестирована, тестирование показало, что все функции работают корректно.

Для примера рассмотрим реализацию и тестирование функции *isquadresidue(a,n)* (в списке разработанных функций находится под номером 11).

М-файл-функция с именем *isquadresidue.m* имеет вид:

```
function f = isquadresidue(a,n)
%% isquadresidue: function, which shows whether a
%    number is a quadratic residue a given modulo n.
%
% Example:
%   f = isquadresidue(3,13);
%  returns -> f = 1;
%
%   f = isquadresidue(3,12);
%   returns -> f = 0;
%
% Author: Vladyslav Borodavka
%    ( v.v.borodavka@hotmail.com )
%
%% calculating whether a number is a quadratic residue %%a
given modulo and input error check:
if nargin~=2
error('Must have exactly two argument')
end
% if the numbers a and m are not relatively prime, then % f = 0
if gcd(a,n) > 1
f = 0;
else
% if n is prime –> calculate Legendre symbol
if isprime(n) == 1
      d = legendresymb(a,n);
          if d == 1
f = 1;
          else
f = 0;
          end
% else  –> calculate Jacobi symbol
else
          d = jacobisymb(a,n);
          if d == -1
f = 0;
          else
if d == 1
                    fprintf('a may be quadratic residue modulo n
                    with probability 1/2');
          end
          end
end
end
```

Входными параметрами данной функции являются целые числа a и n. В теле данной функции используются функция *legendresymb(a,p)* и *jacobisymb(a,n)*, которые также разработаны для теоретико-числового тулбокса.

Результаты тестирования рассматриваемой функции представлены в табл. 5.

Таблица 5

Результаты тестирования функции *isquadresidue*(a,n)

№	Входные данные	Количество разрядов входных данных	Результат
1	3, 13	0, 1	1
2	861, 8191	2, 3	1
3	131059, 131071	5, 5	1
4	21474812, 21474836	7, 7	0
5	2^{11}, $2^{12}+1$	11, 12	*a* may be quadratic residue modulo n with probability 1/2
6	2^{15}, $2^{15}+2^{12}$	15, 15	0
7	2^{50}, $2^{50}+17$	50, 50	*a* may be quadratic residue modulo n with probability 1/2
8	2^{53}, $2^{53}+17$	53, 53	0
9	2^{53}, $2^{54}+17$	53, 54	Warning: Inputs contain values larger than the largest consecutive flint. Result may be inaccurate. 0

В последней колонке выходное значение, равное 1, соответствует положительному ответу на вопрос, является ли число *a* квадратичным вычетом по модулю n, а значение, равное 0, – отрицательному ответу на этот вопрос.

Было установлено, что 53 – это максимальное количество разрядов для входных значений обоих

входных параметров. При разрядности входных даннях, большей 54, данная функция будет отображать результат, но значения могут быть некорректными, о чем свидетельствует выдаваемое предупреждение (последняя строка таблицы).

Заключение

В результате сравнительного анализа наиболее популярных систем компьютерной математики (Matlab, Mathematica, Maple, Mathcad) с точки зрения решения теоретико-числовых задач и вычисления функций элементарной теории чисел было установлено, что наибольшими возможностями обладают системы Mathematica и Maple, содержащие функции для решения базовых задач элементарной теории чисел, находящих применение в криптологии. Возможности систем Matlab и Mathcad представлены функциями: факторизации, проверки простоты чисел, формирования списка простых чисел, меньших заданного, вычисления остатка данного числа по модулю, а также определения наибольшего общего делителя и наименьшего общего кратного двух чисел. В рамках СКМ Mathematica и Maple, в отличие от Matlab и Mathcad, имеется возможность вычислять функцию Эйлера и функцию Мёбиуса, символ Лежандра и символ Якоби, а также значение показателя числа по заданному модулю, решать линейные и квадратичные сравнения (если они совместны), а также системы линейных сравнений с помощью китайской теоремы об остатках и др.

На основании перечисленных встроенных функций системы Matlab было разработано 19 функций, совокупность которых вместе с упомянутыми функциями системы Matlab можно рассматривать в качестве теоретико-числового тулбокса Number Theory Toolbox. Все разработанные функции были протестированы на предмет корректности их реализации. Для каждой из них установлено максимальное число разрядов операндов, с которыми разработанные функции позволяют оперировать.

Разработанный набор функций может быть внедрен как отдельный Toolbox в СКМ Matlab для использования как в учебных, так и научно-исследовательских целях для оптимизации рабочего процесса и ускорения работы с вычислениями.

Список литературы

1. *Таранчук, В.Б. Основные функции систем компьютерной алгебры: учеб, пособие [Текст] / В.Б. Таранчук. – Минск: БГУ, 2013. – 59 с.*
2. *Шишков, М.Л. Системы компьютерной математики как базовый инструмент обучения алгоритмизации и программированию [Текст] / М.Л. Шишков, Т.А. Трохова // Компьютерные инструменты и образование. – 2005. – № 4. – С. 25–34.*
3. *Лысенко И.В. Анализ возможностей решения задач оптимизации средствами систем компьютерной математики [Текст] / И.В. Лысенко, В.О. Бутенко // Системи обробки інформації. – Х: ХУПС, 2016. – Вип. 5(142). – С. 133–136.*
4. *Курбатова Е.А. MATLAB 7. Самоучитель [Текст] / Е.А. Курбатова, 2006. – 256 с.*
5. *Overview of the numtheory Package [Электронный ресурс] // Maplesoft. – 2016. – Режим доступа: https://www.maplesoft.com/support/help/Maple/view.aspx?path=numtheory – 05.02.2017 г.*
6. *Бусыгин Н.Ю. Решение задач в среде Mathcad [Электронный ресурс] / Николай Юрьевич Бусыгин // СПГУТД – Режим доступа: http://eco.sutd.ru/mathcad/index.htm – 06.02.2017 г.*
7. *Number Theoretic Functions [Электронный ресурс] // Wolfram Language & System Documentation Center – Режим доступа: http://reference.wolfram.com/ language/guide/NumberTheoreticFunctions.html– 10.02.2017 г.*
8. *Бедратюк, Л.П. Использование системы компьютерной алгебры MAPLE в элементарной теории чисел / Л.П. Бедратюк, Г.И. Бедратюк // Восточно-Европейский журнал передовых технологий. – 2013. – №6. – С. 10–13.*
9. *Тилборг, ван Х.К.А. Основы криптологии. Профессиональное руководство и интерактивный учебник / Х.К.А. ван Тилборг. – М.: Мир, 2006. – 471 с.*

Надійшла до редколегії 17.02.2017

Рецензент: д-р техн. наук, проф. В.С. Харченко, Національний аерокосмічний університет ім. М.Є. Жуковського «ХАІ», Харків.

РОЗРОБКА ТЕОРЕТИКО-ЧИСЛОВОГО ТУЛБОКСА ДЛЯ СИСТЕМИ КОМП'ЮТЕРНОЇ МАТЕМАТИКИ MATLAB

І.В. Лисенко, В.В. Бородавка

Представлено результати порівняльного аналізу найбільш популярних систем комп'ютерної математики (Matlab, Mathematica, Maple, Mathcad) щодо можливості рішення за їх допомогою теоретико-числових задач і обчислення функцій елементарної теорії чисел. Описано можливості розробленого теоретико-числового тул боксу Number Theory Toolbox для системи Matlab, який представлений множиною функцій для виконання теоретико-числових перетворень, що відносяться до прикладної криптології.

Ключеві слова: *Matlab, функції теорії чисел, теоретико-числовий тулбокс.*

DEVELOPMENT OF NUMBER-THEORETIC TOOLBOXFOR THE COMPUTER MATHEMATICS SYSTEM MATLAB

I.V. Lysenko, V.V. Borodavka

Presents the results of the comparative analysis of the most popular systems of computer mathematics (Matlab, Mathematica, Maple, Mathcad) for possible solutions in the framework of their number-theoretic problems and computation functions of elementary number theory. Describes the possibilities the developed theoretical and numerical toolbox Number Theory Toolbox for system Matlab, represented by a set of functions to perform number-theoretic transformations relating to for applied of cryptology.

Keywords: *Matlab, functions of number theory, number-theoretic toolbox.*