МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені І.І.МЕЧНИКОВА

(повне найменування вищого навчального закладу)

Інститут математики, економіки та механіки

(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерної алгебри та дискретної математики

Дипломна робота

бакалавра

(освітньо-кваліфікаційний рівень)

на тему «Генерування великих простих чисел»
Generating of large primes
Генерирование больших простых чисел

Виконала: студентка 4 курсу, групи 1
напряму підготовки (спеціальності)
6.050102 – Комп'ютерна інженерія

(шифр і назва напряму підготовки, спеціальності)

Король Ю. В.
(прізвище та ініціали)

Керівник Савастру О. В.
(прізвище та ініціали)

Рецензент Бєлозьоров Г. С.
(прізвище та ініціали)

Рекомендовано до захисту:
Протокол засідання кафедри
№_____ від «___»_____р.

Завідувач кафедри
_____ П. Д. Варбанець
(підпис) (прізвище, ініціали)

Захищено на засіданні ДЕК № ___
протокол №___ від «___» _____р.
Оцінка _____
(за 4-х бальною шкалою, за шкалою ECTS, бал.)

Голова ДЕК
_____ А.А. Кобозева
(підпис) (прізвище, ініціали)

Одеса - 2016

АНОТАЦІЯ

В дипломній роботі розробляється тема «Генерування великих простих чисел». В роботі представлені основні алгоритми перевірки чисел на простоту. Пошук великих простих чисел має важливе значення для математики і не тільки. Наприклад, в криптографії великі прості числа використовуються в алгоритмах шифрування з відкритим ключем.

У роботі продемонстровано різні способи пошуку (генерації) простих чисел, тест перевірки на простоту Міллера-Рабіна, алгоритм сильно простого числа, алгоритм Гордона, алгоритм Люка, їх класифікація, з використанням обчислювальних ресурсів сучасних комп’ютерів, зокрема комп’ютерної програми власного авторства.

Метою даного дипломного проекту є – вивчення існуючих алгоритмів генерації великих простих чисел і принципів їх побудови; також реалізація розробленого алгоритму у вигляді комп'ютерної програми.

АННОТАЦИЯ

В дипломной работе разрабатывается тема «Генерация больших простых чисел». В работе представлены основные алгоритмы проверки чисел на простоту. Поиск больших простых чисел имеет важное значение для математики и не только. Например, в криптографии большие простые числа используются в алгоритмах шифрования с открытым ключом.

В работе продемонстрированы различные способы поиска (генерации) простых чисел, тест проверки на простоту Миллера-Рабина, алгоритм сильно простого числа, алгоритм Гордона, алгоритм Люка, их классификация, с использованием вычисляемых ресурсов современных компьютеров, в частности компьютерной программы собственного авторства.

Целью данного дипломного проекта является – изучение существующих алгоритмов генерации больших простых чисел и принципов их построения; также реализация разработанного алгоритма в виде компьютерной программы.

ANNOTATION

In the thesis work developed the theme of "Generation of large primes". The paper presents the basic algorithms of the check numbers in question-the Thoth. The search for large primes is essential to mathematics and not only. For example, in cryptography and large primes are used in encryption algorithms with a public key.

The work demonstrates various ways of searching (generating) Prime numbers, namely, verification test for a simple Miller-Rabin, algorithm is much simple, the Gordon algorithm, the algorithm of the Hatch, their classification, using the computing resources of modern computers, in particular computer Pro-grams of their own creation.

The purpose of this graduation project is – the study of existing algorithms for generating large primes and the principles of their construction; also the implementation of the developed algorithm in the form of a computer program.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ ТА ТЕРМІНІВ

ЕЛЕМЕНТ ∈ - належить. Якщо a – елемент множини A, то пишуть a ∈ A й читають "a належить A". Якщо a не є елементом множини A, пишуть a ∉ A й читають "не належить A". Спочатку відносини "міститься" і "належить" ("є елементом") не розрізняли, але з часом ці поняття вимагають розмежування.

Елементи $\subset, \supset$ (міститься, містить). Якщо A і B – дві множини і в A немає елементів, які не належать, то кажуть що міститься в B. Пишуть A⊂B⊃A (містить A).

Знак ≡ (тотожність). Тотожність – рівність двох аналітичних виразів, справедливий для будь-яких допустимих значень вхідних у нього букв. Рівність a+b = b+a справедливо при всіх числових значеннях a і b, і тому є тотожністю.

≡ (mod m) (порівнянність). Порівняння – співвідношення між двома цілими числами n і m, що означає, що різниця n–m цих чисел ділиться на задане ціле число a, що називається модулем порівняння; пишеться: n≡m (mod a) і читається "числа n і m порівнянні з модулю a".

∩ (перетин). Перетин множин – це множина, якій належать ті і тільки ті елементи, які одночасно належать усім даними множиною.

∅ (порожня множина). Множина, що не містить жодного елементу.

⊇ (надмножина). A⊇B означає «кожен елемент із A також являється елементом із B».

ℤ (цілі числа). ℤ означає множину {…-3, -2, -1, 0, 1, 2, 3…}.

| | (модуль). |x| означає абсолютну величину x. |A| означає потужність множини A і дорівнює, якщо звичайно, числа елементів A.

ℕ (натуральні числа). ℕ означає множину {1, 2, 3, …} або рідше {0, 1, 2, 3, …} (залежно від ситуації).

$\prod$ (множення). $\prod_{k=1}^{n} q_k$ означає «множення q_k для всіх k від 1 до n», тобто $q_1 * q_2 * \ldots * q_n$.

√ - арифметичний квадратний корень.

$\sqrt{x}$ - означає невід'ємне дійсне число, яке в квадраті дає x.

| - дільник, ділить націло.

⌈ ⌉ - Огруглення числа до цілого в більшу сторону.

⌊ ⌋ - Огруглення числа до цілого в меншу сторону.

{ } - усередині дужок записуються елементи множини.

± (плюс-минус) – з точністю.

π (число Пи). Математична константа, рівна відношенню довжини окружності до її діаметру. $\pi \approx 3{,}14159265$.

<> - середнє значення, усереднення.

∞ - бескінечність. Елемент розширеної числової прямої, який більше любого числа.

≪ - набагато менше.

≤ - менше або дорівнює.

< - менше.

≥ - більше або дорівнює.

> - більше.

≠ - нерівність.

ln - символ натурального логарифма.

Ф - функція Ейлера φ(n) в теорії чисел, а також позначення довільної функції.

Max – максимум.

q̂- позначення елементу.

ЕОМ (електронна обчислювальна машина) - загальна назва для обчислювальних машин, що є електронними на відміну від електромеханічних та механічних обчислювальних машин.

ВВЕДЕННЯ

Актуальність роботи полягає в тому, що досі не знайдено справедливої математичної формули, яка би точно (при всіх допустимих значеннях аргументів) дозволяла знайти як завгодно велике просте число. А тому саме проблема знаходження найоптимальнішого способу їх генерації являється однією з небагатьох значних «білих плям» не тільки у теорії чисел, а у математиці загалом.

Постановка задачі: розгляд алгоритмів побудови великих простих чисел, тестування на простоту великих простих чисел, програмна реалізація тесту Міллера-Рабіна, розбиття Гольдбаха та нового алгоритму GPGA.

Предметна область: пошук великих простих чисел.

Об’єкт: прості числа.

Мета: поглибити знання в зазначеній області, розглянути основні способи генерації простих чисел та вибрати з них найоптимальніший, найефективніший, застосувавши ці методи експериментально, ознайомлення з методами генерування великих простих чисел, опис, аналіз та програмна реалізація .

Виділення простих чисел є складним завданням математики. Вчені протягом багатьох століть намагаються знайти закон, по якому можна визначити n-е просте число. Перший, хто займався цією проблемою, був великий математик давнини Ератосфен.

Першу відому нам таблицю простих чисел склав італійський математик П'єтро Антоніо Катальді в 1603 р. Вона захоплювала всі прості числа від 2 до 743.

У 1770 р. Німецький математик Йоганн Генріх Ламберт опублікував таблицю найменших дільників чисел, не переважаючих 102000 і які не діляться

на 2, 3, 5. Першим значним успіхом у визначенні простоти цілих чисел з'явилися теорема Вільсона і теорема Ферма, які є центральними в даній проблемі донині..Вже на протязі багатьох століть видатні математики поступово розробляли й пояснювали цю, на перший погляд, загальновідому й зрозумілу теорію, пов’язану з простими числами, проте навіть зараз, через стільки часу, деякі властивості й феномени простих чисел, зокрема їх хаотичний розподіл серед натуральних чисел, досі не мають належного пояснення. Окремі видатні вчені-математики присвячували їм ціле життя.

В дипломній роботі представлені основні алгоритми перевірки чисел на простоту. Чому ж так важливо знати просте число чи ні? Пошук великих простих чисел має важливе значення для математики і не тільки. Наприклад, в криптографії великі прості числа використовуються в алгоритмах шифрування з відкритим ключем.

Також у роботі продемонстровано різні способи пошуку (генерації) простих чисел, тест перевірки на простоту Міллера-Рабіна, алгоритм сильно простого числа, алгоритм Гордона, алгоритм Люка, їх класифікація, з використанням обчислювальних ресурсів сучасних комп’ютерів, зокрема комп’ютерної програми власного авторства.

ВИСНОВОК

Як вже було розглянуто у вступі й основній частині, головна мета цієї роботи полягала в ознайомленні з методами тестування на простоту та методами генерування великих простих чисел, а також у пошуку найбільш раціонального способу їх знаходження.

В роботі були розглянуті різні види й методи перевірки на простоту великих простих чисел, спеціальні форми їх запису, тести простоти, їх різновиди. Деякі з них були розглянуті більш детально.

Детально розглядався алгоритм перевірки на простоту методом Мілера-Рабіна та новий алгоритм генерації Гольдбаха (GPGA). Були проведені обчислювальні експерименти. Спеціально була створена комп’ютерна програма, яка реалізувала ці алгоритми.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Miller G. Riemann’s hypothesis and tests for primality // Journal of Computer and System Sciences._ 1976._ Vol. 13, no. 3._ Pp. 300–317.

2. Rabin M. Probabilistic algorithm for testing primality // Journal of Number Theory._ 1980._ Vol. 12, no. 1._ Pp. 128–138.

3. Гашков С. Спрощене обгрунтування ймовірного тесту Міллера-Рабіна для перевірки простоти чисел // Дискретна математика. — 1998. — Т. 10, № 4. — С. 35–38.

4. Mihailescu P. Fast generation of provable primes using search in arithmetic progressions // Advances in Cryptology – CRYPTO ’94. — Vol. 839 Of Lecture Notes Of Computer Science. — Springer, 94. — Pp. 282–293.

5. Williams H., Schmid B. Some remarks cocerning the MIT public-key cryptosystem // BIT. — 1979. — Vol. 19. — Pp. 525–538.

6. Галочкин А., Нестеренко Ю., Шидловский А. Введення в теорію чисел. — М.: Изд-во Московского Університету, 1984. — С. 152.
7. Gordon J. Strong RSA keys // Electronic Letters. — 1984. — Vol. 20, no. 12. — Pp. 514–516. — June 1984.

8. S. Kak, Goldbach partitions and sequences. Resonance 19: 1028-1037, 2014.

9. Generating primes using partitions [Електронний ресурс] – Режим доступу: http://arxiv.org/abs/1505.00253