UNIVERSITAS INDONESIA

KUNCI RAHASIA BERDASARKAN POLINOMIAL CHEBYSHEV PADA PROSES ENKRIPSI DAN DEKRIPSI

TESIS

RAJA LENI MURZAINI
0906577381

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
PROGRAM MAGISTER MATEMATIKA
DEPOK
JUNI 2011

UNIVERSITAS INDONESIA

KUNCI RAHASIA BERDASARKAN POLINOMIAL CHEBYSHEV PADA PROSES ENKRIPSI DAN DEKRIPSI

TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar Magister Sains

RAJA LENI MURZAINI
0906577381

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
PROGRAM MAGISTER MATEMATIKA
DEPOK
JUNI 2011

HALAMAN PERNYATAAN ORISINALITAS

Tesis ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.

Nama : RAJA LENI MURZAINI
NPM : 0906577381

Tanda Tangan : .....

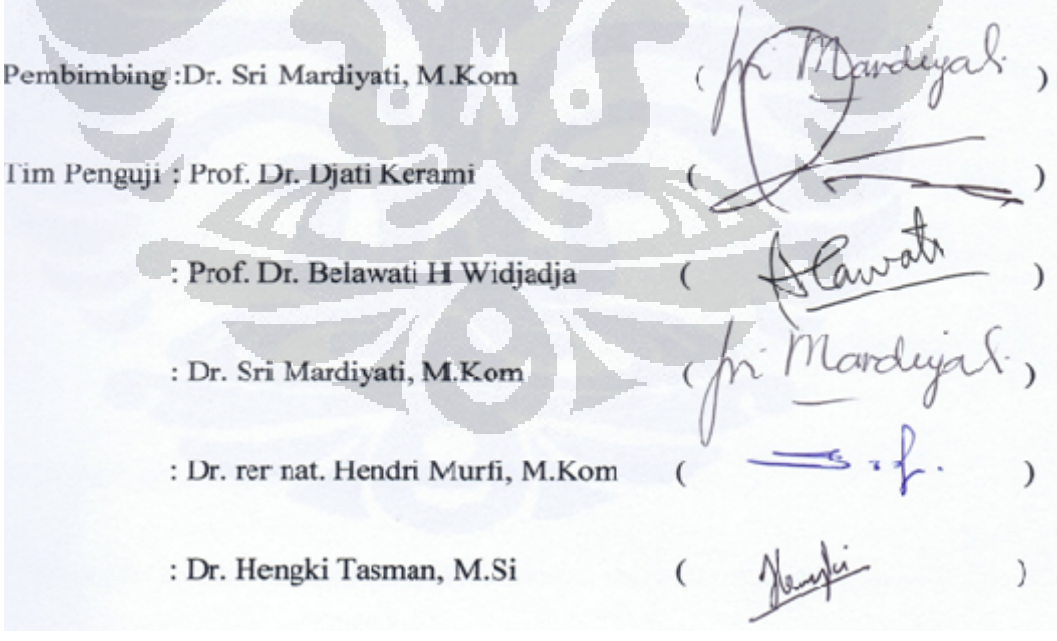
Tanggal : 27 Juni 2011

HALAMAN PENGESAHAN

Tesis ini diajukan oleh :
Nama : Raja Leni Murzaini
NPM : 0906577381
Program Studi : Magister Matematika
Judul Tesis :
Kunci Rahasia Berdasarkan Polinomial Chebyshev pada Proses Enkripsi dan Dekripsi

Telah berhasil dipertahankan dihadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Sains pada Program Studi Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia.

DEWAN PENGUJI



Ditetapkan di Depok

Tanggal : 27 Juni 2011

KATA PENGANTAR/UCAPAN TERIMA KASIH

Alhamdulillah. Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa. Atas berkat dan rahmatNya, penulis dapat merampungkan tesis yang berjudul **Kunci Rahasia Berdasarkan Polinomial Chebyshev pada Proses Enkripsi dan Dekripsi**. Penulisan tesis ini dilakukan dalam rangka memenuhi salah satu syarat untuk meperoleh gelar Magister Sains Program Studi Magister Matematika pada Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Indonesia.

Perkembangan teknologi telah memungkinkan seseorang mengirimkan sejumlah informasi kepada pihak lain secara *on-line*. Kriptografi adalah suatu teknik merahasiakan informasi. Tesis ini membahas tentang kriptografi. Penelitian dalam tesis ini adalah kriptografi yang menggunakan kunci asimetri.

Penulis banyak memperoleh bantuan dan dukungan dari berbagai pihak dalam merampungkan tesis ini. Untuk itu penulis menghaturkan terima kasih kepada:

1. Dr. Sri Mardiyati, M.Kom. selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penulis dalam penyusunan tesis ini;
2. Prof. Dr. Djati Kerami selaku ketua program studi Magister Matematika sekaligus dosen penasehat akademik penulis;
3. Suami, orang tua, dan keluarga penulis yang telah memberikan dukungan material dan spiritual;
4. Segenap bapak-bapak dan ibu-ibu dosen didepartemen Matematika pada Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Indonesia atas bimbingan selama penulis menempuh pendidikan;
5. Para sahabat atas saran yang telah diberikan.

Penulis berharap semoga Tuhan Yang Maha Esa membalas segala kebaikan semua pihak. Akhir kata, semoga tesis ini membawa manfaat bagi pengembangan ilmu pengetahuan.

Depok, Juni 2011

Raja Leni Murzaini

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

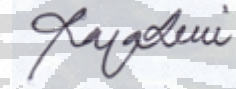
Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama	: Raja Leni Murzaini
NPM	: 0906577381
Program Studi	: Magister Matematika
Departemen	: Matematika
Fakultas	: Matematika dan Ilmu Pengetahuan Alam
Jenis karya	: Tesis

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :
Kunci Rahasia Berdasarkan Polinomial Chebyshev pada Proses Enkripsi dan Dekripsi
beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di: Depok
Pada tanggal: 27 Juni 2011
Yang menyatakan,



(Raja Leni Murzaini)

ABSTRAK

Nama : Raja Leni Murzaini
Program Studi : Magister Matematika
Judul :
Kunci Rahasia Berdasarkan Polinomial Chebyshev pada Proses Enkripsi dan Dekripsi

Algoritma Diffie-Hellman digunakan dalam pembentukan kunci rahasia yang berdasarkan polinomial Chebyshev. Kemudian kunci rahasia tersebut digunakan pada proses enkripsi dan dekripsi.

Kata kunci:
Algoritma Diffie-Hellman, polinomial Chebyshev, kunci rahasia, enkripsi, dan dekripsi.

ABSTRACT

Name : Raja Leni Murzaini
Study Program : Master Program on Mathematics
Title :
Secret Key Based on Chebyshev Polynomial For Encryption and Decryption Process

Diffie-Hellman algorithm is used in generating the secret key based on Chebyshev polynomials. Then the secret key is used for encryption and decryption process.

Keywords:
Diffie-Hellman algorithm, Chebyshev polynomials, secret key, encryption, and decryption.

DAFTAR ISI

HALAMAN JUDUL ……..…….. i
HALAMAN PERNYATAAN ORISINALITAS ……. ii
LEMBAR PENGESAHAN …….. iii
KATA PENGANTAR/UCAPAN TERIMA KASIH …….. iv
LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH ……..……... v
ABSTRAK ……... vi
ABSTRACT ……..... vii
DAFTAR ISI …… viii
DAFTAR GAMBAR ……... ix
DAFTAR TABEL …… x
DAFTAR LAMPIRAN …… xi

1. PENDAHULUAN ……..…. 1
1.1. Latar Belakang ……..…… 1
1.2. Perumusan Masalah ……..…… 3
1.3. Tujuan Penelitian ……..…… 3
1.4. Batasan Penelitian …… 3
1.5. Sistematika Penulisan ……... 3

2. LANDASAN TEORI ……..…..... 4
2.1. Pembangkitan Kunci pada Algoritma Diffie-Hellman ……. 4
2.2. Polinomial Chebyshev …….. 5
2.3. Lapangan Hingga $\mathbb{Z}_p$ …… 8
2.4. Polinomial Chebyshev atas Lapangan Hingga $\mathbb{Z}_p$ ……..…… 11

3. KUNCI RAHASIA BERDASARKAN POLINOMIAL CHEBYSHEV PADA PROSES ENKRPSI DAN DEKRIPSI 12
3.1. Pembangkitan Kunci Berdasarkan Polinomial Chebyshev pada Algoritma Diffie-Hellman ……..……..... 12
3.2. Proses Enkripsi dan Dekripsi …….. 14
3.3. Algoritma-Algoritma yang Diperlukan dalam Simulasi Pembentukan Kunci, Enkripsi, dan Dekripsi ……. 15
3.4. Beberapa Hasil Uji Coba Setelah Algoritma-Algoritma Diimplemetasikan dalam Bahasa Pemrogramam Matlab ……..... 18

4. KESIMPULAN DAN SARAN ……..... 21

DAFTAR REFERENSI ……..….... 22

LAMPIRAN-LAMPIRAN ……. 23

DAFTAR GAMBAR

Gambar 1.1. Penjelasan sederhana mengenai proses enkripsi dan dekripsi …...… 2

DAFTAR TABEL

Tabel 3.1. Penjumlahan untuk $\mathbb{Z}_7$ ……..……..……...… 10

Tabel 3.2. Perkalian untuk $\mathbb{Z}_7$ ……..…… 11

DAFTAR LAMPIRAN

Lampiran 1. Program Simulasi Pembangkitan Kunci ……..…… 24
Lampiran 2. Program Simulasi Proses Enkripsi/Dekripsi ..……..…… 26
Lampiran 3. Program Algoritma Euclide yang Diperluas ……..…….. 27
Lampiran 4. Progam Menentukan *Invers* Perkalian Modulo p ……..…….. 28
Lampiran 5. Program Menu Utama ……..……..……... 29

BAB 1
PENDAHULUAN

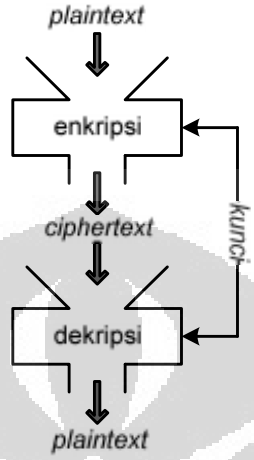
1.1. LATAR BELAKANG

Perkembangan teknologi telah memungkinkan seseorang mengirimkan sejumlah informasi kepada pihak lain secara *on-line*. Salah satu contoh informasi yang dikirimkan adalah nomor kartu kredit dalam pembayaran secara *non tunai*. Apabila informasi tersebut jatuh kepada orang-orang yang tidak berhak mengaksesnya, maka tentu saja hal ini akan merugikan si pemilik kartu kredit tersebut. Untuk itu diperlukan suatu teknik guna mengamankan informasi tersebut. Contoh lain adalah bila sesorang bermaksud melakukan transaksi melalui Anjungan Tunai Mandiri (ATM). Orang tersebut memasukkan kartu ATM beserta *Personal Identification Number* (PIN) ke dalam mesin ATM. Sistem pada mesin ATM mengirim informasi tersebut ke komputer bank sentral. Komputer bank sentral merespon informasi tersebut. Kemudian mengirimkan kembali informasi, apakah transaksi dapat dilanjutkan atau ditolak. Diperlukan proteksi dalam mengirim informasi tersebut (Piper & Murphy, 2002).

Kriptografi berasal dari bahasa Yunani yaitu *kryptos* yang berarti rahasia dan *graphin* yang berarti tulisan. Berdasarkan Kamus Besar Bahasa Indonesia Edisi Ketiga yang diterbitkan oleh Balai Pustaka, kriptografi dimaknai sebagai teknik yang mengubah data menjadi berbeda dari aslinya dengan menggunakan algoritma matematika sedemikian sehingga orang yang tidak mengetahui kuncinya tidak dapat membongkar data tersebut. Sedangkan algoritma didefinisikan sebagai sejumlah perintah yang disusun secara sistematis guna menyelesaikan masalah.

Terdapat sejumlah istilah dalam kriptografi yaitu pengirim, penerima, *plaintext*, enkripsi, *ciphertext*, dekripsi, kunci, kriptanalisis, dan penyerang. Pengirim adalah entitas yang mengirim informasi. Penerima adalah entitas yang menerima informasi. Enkripsi adalah suatu proses yang mentransformasikan informasi yang hendak dikirim (*plaintext*) menjadi suatu informasi yang tidak dapat dimengerti maknanya (*ciphertext*). *Ciphertext* inilah yang dikirirm kepada pihak penerima. Sedangkan dekripsi merupakan kebalikan dari enkripsi, yaitu

suatu proses yang mengembalikan *ciphertext* menjadi *plaintext*. Dalam proses enkripsi maupun dekripsi, keduanya memerlukan bantuan yang disebut kunci. Gambar 1.1. merupakan penjelasan sederhana mengenai proses enkripsi dan dekripsi.



Gambar 1.1 Penjelasan sederhana mengenai proses enkripsi dan dekripsi

Kriptanalisis adalah ilmu memecahkan *ciphertext* menjadi *plaintext* tanpa mengetahui kuncinya secara wajar. Pelaku kriptanalisis disebut penyerang. Kekuatan suatu algoritma dan kerahasiaan kunci menentukan keamanan suatu *ciphertext* (Zimmermann, 2004).

Kunci yang digunakan dalam kriptografi dibedakan dalam dua jenis yaitu kunci simetri dan kunci asimetri. Jika kunci yang sama digunakan pada enkripsi dan dekripsi maka disebut kunci simetri. Beberapa contoh algoritma enkripsi dan dekripsi yang menggunakan kunci simetri adalah *Data Encryption Standard* (DES), *International Data Encryption Algorithm* (IDEA), dan *Advanced Encryption Standard* (AES) (Menezes, van Oorschot, & Vanstone, 1997).

Jika kunci yang berbeda digunakan pada proses enkripsi dan dekripsi maka disebut kunci asimetri. Kunci publik boleh diketahui oleh semua orang, namun tidak demikian halnya dengan kunci privat dan kunci rahasia. Kunci-kunci ini saling berhubungan. Beberapa contoh algoritma enkripsi dan dekripsi yang menggunakan kunci asimetri adalah algoritma Rivest, Shamir, dan Adleman (RSA), algoritma El Gamal, dan *Digital Signature Algorithm* (DSA) (Zimmermann, 2004).

Kunci publik biasanya dibentuk berdasarkan suatu fungsi matematika. Sebagai contoh, untuk algoritma RSA, algoritma El Gamal, dan DSA, pembentukan kunci publik menggunakan algoritma Diffie-Hellman. Algoritma Diffie-Hellman menggunakan monomial x^n dalam pembentukan kunci (Juari, 2006).

1.2. PERUMUSAN MASALAH

Bagaimanakah bila monomial x^n pada pembentukan kunci publik pada algoritma Diffie-Hellman diganti dengan polinomial Chebyshev?

1.3. TUJUAN PENULISAN

- Membentuk kunci publik berdasarkan polinomial Chebyshev pada algoritma Diffie-Hellman.
- Memberikan simulasi proses enkripsi dan dekripsi yang bersesuaian.

1.4. BATASAN MASALAH

Batasan masalah dalam penulisan tugas akhir ini adalah menggunakan polinomial Chebyshev jenis pertama.

1.5. SISTEMATIKA PENULISAN

Bab 1 membahas tentang latar belakang, perumusan masalah, tujuan penulisan, batasan masalah, dan sistematika penulisan.

Bab 2 membahas tentang landasan teori mengenai pembangkitan kunci pada algoritma Diffie-Hellman, polinomial Chebyshev, lapangan hingga $\mathbb{Z}_p$, dan polinomial Chebyshev atas lapangan hingga $\mathbb{Z}_p$.

Bab 3 membahas tentang pembangkitan kunci berdasarkan polinomial Chebyshev pada algoritma Diffie-Hellman, enkripsi, dan dekripsi, algoritma-algoritma yang diperlukan dalam simulasi pembangkitkan kunci, enkripsi dan dekripsi, beserta hasil uji coba setelah algoritma-algoritma tersebut diimplemetasikan dalam bahasa pemrograman Matlab.

Bab 4 membahas tentang kesimpulan dan saran yang berkaitan dengan penelitian.

BAB 2
LANDASAN TEORI

Bab ini mengulas tentang pembangkitan kunci pada algoritma Diffie-Hellman, polinomial Chebyshev, lapangan hingga $\mathbb{Z}_p$, dan polinomial Chebyshev atas lapangan hingga $\mathbb{Z}_p$ dengan p adalah bilangan prima.

2.1. PEMBANGKITAN KUNCI PADA ALGORITMA DIFFIE-HELLMAN

Pada bagian ini akan dibahas algoritma pertukaran kunci Diffie-Hellman. Algoritma Diffie-Hellman adalah suatu algoritma yang menggunakan monomial x^n dalam pembangkitan kunci. Algoritma Diffie-Hellman pertama kali diperkenalkan oleh Whitfield Diffie dan Martin Hellman pada tahun 1976.

Jika Bob bermaksud mengirim informasi rahasia kepada Alice maka keduanya harus sepakat dengan satu kunci rahasia. Untuk itu mereka sepakat menggunakan algoritma Diffie-Hellman. Langkah-langkah yang dilakukan adalah:

1. Alice menentukan bilangan prima p dan bilangan bulat x dengan $1 < x < p$.
2. Alice memilih suatu bilangan bulat m dengan $1 < m < p$.
3. Alice menghitung $e = x^m \bmod p$.
4. Alice mengirim p, x, dan e kepada Bob.
5. Bob memilih bilangan bulat n dengan $1 < n < p$.
6. Bob menghitung $b = x^n \bmod p$.
7. Bob mengirim b kepada Alice.
8. Alice menghitung kunci rahasia $c = b^m \bmod p$.
9. Bob menghitung kunci rahasia $d = e^n \bmod p$ (Fee & Monagan, 2005).

Kunci rahasia yang mereka bangkitkan adalah sama, karena

$$\begin{aligned} c &= b^m \bmod p, \qquad b = x^n \bmod p \\ &= (x^n \bmod p)^m \bmod p \\ &= \underbrace{[(x^n \bmod p)(x^n \bmod p) \dots (x^n \bmod p)]}_{m\ faktor} \bmod p \\ &= (x^n)^m \bmod p \end{aligned}$$

$$= (x^m)^n \bmod p$$
$$= (x^m \bmod p)^n \bmod p$$
$$= e^n \bmod p$$
$$= d$$

2.2. POLINOMIAL CHEBYSHEV

Pada bagian ini berturut-turut akan diuraikan mengenai definisi fungsi, polinomial, polinomial Chebyshev, relasi rekursif pada polinomial Chebyshev dan sifat komutatif polinomial Chebyshev di bawah operasi komposisi fungsi.

Pafnuty Lvovich Chebyshev adalah salah seorang matematikawan Rusia yang hidup pada abad 19. Beliau adalah penemu salah satu bentuk polinomial, yang kemudian dikenal dengan nama polinomial Chebyshev.

Polinomial Chebyshev merupakan salah satu bentuk fungsi. Karenanya pembahasan pada bagian ini dimulai dengan definisi fungsi.

Definisi 2.1. Misalkan diketahui dua himpunan A dan B yang masing-masing bukan merupakan himpunan kosong. Suatu **fungsi** (**pemetaan**) f dari A ke B adalah suatu aturan yang memasangkan setiap elemen di A dengan tepat satu elemen di B (Clapham, 1996).

Salah satu bentuk khusus fungsi adalah polinomial yang definisinya diuraikan di bawah ini.

Definisi 2.2. Misalkan $a_i \in \mathbb{R}$, dengan tidak semua a_i bernilai nol, untuk $i = 0, 1, \ldots, n$. Maka suatu fungsi p dari $\mathbb{R}$ ke $\mathbb{R}$ yang didefinisikan sebagai

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0$$

disebut **polinomial** dalam variabel x. Jika $a_n \neq 0$ maka $p(x)$ dikatakan berderajat n dan bilangan a_i disebut koefisien dari suku ke-i, $i = 0, 1, \ldots, n$. (Clapham, 1996).

Salah satu bentuk khusus polinomial adalah polinomial Chebyshev yang didefinisikan sebagai berikut.

Definisi 2.3. Polinomial Chebyshev $T_n(x)$ jenis pertama adalah suatu polinomial dalam variabel x dan berderajat n, yang memenuhi

$$T_n(x) = \cos(n \arccos x)$$

dengan $-1 \leq x \leq 1$, untuk $x > 1$ persamaan yang digunakan adalah

$$T_n(x) = cosh\,(n\,arccosh\,x)$$

dengan $n = 0, 1, 2, \ldots$ (Mason & Handscomb, 2003).

Pada polinomial-polinomial Chebyhev berlaku relasi rekursif yang ditunjukkan oleh teorema 2.4 berikut.

Teorema 2.4. Polinomial-polinomial Chebyshev memenuhi **relasi rekursif**, yaitu

$$T_n(x) = 2x\,T_{n-1}(x) - T_{n-2}(x) \qquad (2.2.)$$

dengan $T_0(x) = 1, T_1(x) = x,\ -1 \leq x \leq 1,$ dan $n = 2, 3, 4, \ldots$

Bukti:

Diketahui polinomial Chebyshev $T_n(x) = cos\,(n\,arccos\,x)$ dengan

$-1 \leq x \leq 1$, dan $n = 0, 1, 2, \ldots$

Jika dimisalkan $\theta = arccos\,x$ maka $0 \leq \theta \leq \pi$ maka $T_n(x) = cos\,n\theta$.

Untuk $n = 0$ diperoleh $T_0(x) = cos\,0 = 1$.

Untuk $n = 1$ diperoleh $T_1(x) = cos\,\theta = cos\,(arccos\,x) = x.$ (2.1.)

Akan ditunjukkan bahwa

$$T_n(x) = 2x\,T_{n-1}(x) - T_{n-2}(x)$$

Pandang polinomial Chebyshev

$$T_{n-1}(x) = \cos(n-1)\theta$$
$$= cos\,n\theta\,cos\,\theta + sin\,n\theta\,sin\,\theta. \qquad (2.3.)$$

Dari persamaan (2.1.) diketahui bahwa $\cos\theta = x$.

Jika ruas kiri pada persamaan (2.3.) dikalikan dengan $2x$ dan pada ruas kanan persamaan (2.3.) dikalikan dengan $2\,cos\,\theta$ maka diperoleh

$$2x\,T_{n-1}(x) = 2\,cos\,\theta\,(cos\,n\theta\,cos\,\theta + sin\,n\theta\,sin\,\theta)$$
$$= 2\,cos\,n\theta\,cos^2\theta + sin\,n\theta\,sin\,2\theta \qquad (2.4.)$$

Pandang polinomial Chebyshev

$$T_{n-2}(x) = cos(n-2)\theta$$
$$= cos\,n\theta\,cos\,2\theta + sin\,n\theta\,sin\,2\theta \qquad (2.5.)$$

Jika (2.4.) dikurangi dengan (2.5.) maka diperoleh

$$2x\,T_{n-1}(x) - T_{n-2}(x) = 2\,cos\,n\theta\,cos^2\theta - cos\,n\theta\,cos\,2\theta$$
$$= cos\,n\theta\,(2cos^2\theta - (1 - 2sin^2 x))$$

$$= \cos n\theta$$
$$= T_n(x)$$
$$T_n(x) = 2x\,T_{n-1}(x) - T_{n-2}(x). \quad \blacksquare$$

Contoh 2.5. Berdasarkan relasi rekursif pada persamaan (2.2.), maka

untuk $n = 2$ diperoleh $T_2(x) = 2x\,T_1(x) - T_0(x)$
$$= 2x^2 - 1$$
untuk $n = 3$ diperoleh $T_3(x) = 2x\,T_2(x) - T_1(x)$
$$= 2x(2x^2 - 1) - x$$
$$= 4x^3 - 3x$$
untuk $n = 4$ diperoleh $T_4(x) = 2x\,T_3(x) - T_2(x)$
$$= 2x(4x^3 - 3x) - (2x^2 - 1)$$
$$= 8x^4 - 8x^2 + 1$$
untuk $n = 5$ diperoleh $T_5(x) = 2x\,T_4(x) - T_3(x)$
$$= 2x(8x^2 - 8x^2 + 1) - (4x^2 - 3x)$$
$$= 16x^5 - 20x^3 + 5x$$
untuk $n = 6$ diperoleh $T_6(x) = 2x\,T_5(x) - T_4(x)$
$$= 2x(16x^5 - 20x^3 + 5x) - (8x^4 - 8x^2 + 1)$$
$$= 32x^6 - 48x^4 + 18x^2 - 1.$$

Pada polinomial-polinomial Chebyshev berlaku sifat komutatif dibawah operasi komposisi fungsi, yang ditunjukkan oleh teorema 2.6 berikut.

Teorema 2.6. Untuk setiap polinomial Chebyshev $T_m(x)$ dan $T_n(x)$ berlaku sifat **komutatif** dibawah operasi komposisi fungsi yaitu

$$T_n(T_m(x)) = T_m(T_n(x))$$

dengan $m, n \in \mathbb{Z}^+$.

Bukti:

Ambil sebarang polinomial Chebyshev $T_m(x)$ dan $T_n(x)$ dengan $m, n \in \mathbb{Z}^+$.

Akan ditunjukkan bahwa

$$T_n(T_m(x)) = T_m(T_n(x)).$$

Berdasarkan definisi polinomial Chebyshev berarti $T_m(x) = cos(m\,arccos\,x)$, dan

$$\begin{aligned} T_n(T_m(x)) &= cos\,(n\,arccos\,(T_m(x)) \\ &= cos\,(n\,arccos\,(cos\,(m\,arccos\,x)) \\ &= cos\,(nm\,arccos\,x) \\ &= cos\,(mn\,arccos\,x) \\ &= cos\,(m\,arccos\,(cos\,(n\,arccos\,x)) \\ &= cos\,(m\,arccos\,(T_n(x)) \\ &= T_m(T_n(x)). \end{aligned}$$

Untuk $x > 1$ pembuktian $T_n(T_m(x)) = T_m(T_n(x))$ menggunakan persamaan $T_n(x) = cosh\,(n\,arccosh\,x)$. ■

2.3. LAPANGAN HINGGA $\mathbb{Z}_p$

Pada bagian ini akan diuraikan berturut-turut tentang sistem matematika, lapangan, lapangan hingga, dan lapangan hingga $\mathbb{Z}_p$ dengan p adalah bilangan prima. Pembahasan pada bagian ini dimulai dengan sistem matematika.

Misalkan diketahui suatu himpunan tak kosong S. Pada himpunan S didefinisikan pemetaan

$$\circ : S \times S \rightarrow S$$

Peta dari (a, b) dinyatakan dengan $a \circ b$ atau ab. Himpunan S bersama dengan operasi $\circ$ dinamakan sistem matematika, dinyatakan dengan $(S, \circ)$.

Pada himpunan S dapat juga didefinisikan dua operasi, misalnya

$$\circ : S \times S \rightarrow S \text{ dan } * : S \times S \rightarrow S$$

dinyatakan dengan $(S, \circ, *)$ (Arifin , 2001).

Definisi 2.7. Msalkan diketahui himpunan tak kosong F bersama dengan

operasi tambah $+ : F \times F \rightarrow F$

$$+ : (a, b) \mapsto a + b, \text{ dan}$$

operasi kali $\circ : F \times F \rightarrow F$

$$\circ : (a, b) \mapsto ab.$$

Sistem matematika $(F, +, \circ)$ disebut lapangan jika memenuhi sifat-sifat di bawah ini.

1. Terhadap operasi tambah, sistem matematika $(F, +)$ memenuhi hubungan berikut.
   a. $a + b = b + a$ untuk setiap $a, b \in F$ (sifat komutatif).
   b. $(a + b) + c = a + (b + c)$ untuk setiap $a, b, c \in F$ (sifat asosiatif).
   c. Terdapat $0 \in F$ yang bersifat
      $a + 0 = 0 + a = a$ untuk setiap $a \in F$.
      Elemen 0 disebut identitas terhadap operasi tambah.
   d. Untuk setiap $a \in F$ terdapat $-a \in F$ yang bersifat
      $a + (-a) = (-a) + a = 0$.
      Elemen $(-a)$ disebut *invers* terhadap operasi tambah.
2. Terhadap operasi kali, sistem matematika $(F, \circ)$ memenuhi hubungan berikut.
   a. $ab = ba$ untuk setiap $a, b \in F$ (sifat komutatif).
   b. $(ab)c = a(bc)$ untuk setiap $a, b, c \in F$ (sifat asosiatif).
   c. Terdapat $1 \in F$ dengan $1 \neq 0$ yang bersifat
      $a1 = 1a = a$ untuk setiap $a \in F$.
      Elemen 1 disebut identitas terhadap operasi kali.
   d. Untuk setiap $a \in F, a \neq 0$, terdapat $a^{-1} \in F$ yang bersifat
      $aa^{-1} = a^{-1}a = 1$.
      Elemen a^{-1} disebut *invers* terhadap operasi kali.
3. Untuk setiap $a, b, c \in F$ berlaku
   a. $a(b + c) = ab + ac$ (sifat distributif kiri) dan
   b. $(a + b)c = ac + bc$ (sifat distributif kanan).

Berdasarkan banyak elemen pada suatu lapangan F, maka lapangan F terbagi menjadi lapangan tak hingga dan lapangan hingga. Jika lapangan F memuat sebanyak tak hingga elemen, maka lapangan F disebut lapangan tak hingga. Contohnya adalah lapangan $\mathbb{R}$. Jika suatu lapangan F memuat sebanyak hingga elemen, maka lapangan F disebut lapangan hingga (Arifin, 2001). Untuk mencari contoh lapangan hingga, maka terlebih dahulu dibahas pengertian kongruen ke suatu modulo.

Definisi 2.8. Misalkan diketahui $a, b \in \mathbb{Z}$, dan $p \in \mathbb{Z}^+$, jika $p \mid (a - b)$ maka dikatakan ***a* kongruen dengan *b* modulo *p***, dinyatakan dengan $a \equiv b \ (mod\ p)$.

Untuk $p \in \mathbb{Z},\ p > 1$ didefinisikan

$$\mathbb{Z}_p = \{x \bmod p : x \in \mathbb{Z}\} = \{\bar{0}, \bar{1}, \bar{2}, \dots, \overline{p-1}\}.$$

Pada $\mathbb{Z}_p$ didefinisikan operasi:

- Penjumlahan modulo p, yaitu

  $\bar{a} + \bar{b} := \overline{(a + b) \bmod p}$ untuk setiap $\bar{a}, \bar{b} \in \mathbb{Z}_p$.

- Perkalian modulo p, yaitu

  $\bar{a} \,.\, \bar{b} := \overline{(a.b) \bmod p}$ untuk setiap $\bar{a}, \bar{b} \in \mathbb{Z}_p$ (Shoup, 2005).

Teorema 2.9. Misalkan $\mathbb{Z}_p$ adalah himpunan bilangan bulat modulo p, dengan p adalah bilangan prima maka $\mathbb{Z}_p$ adalah suatu lapangan. Karena $\mathbb{Z}_p$ memuat sejumlah berhingga elemen maka $\mathbb{Z}_p$ disebut lapangan hingga (Herstein, 1996).

Contoh 2.10. Pandang $\mathbb{Z}_7 = \{\bar{0}, \bar{1}, \bar{3}, \bar{4}, \bar{5}, \bar{6}\}$. Di bawah ini berturut-turut adalah tabel penjumlahan dan perkalian untuk $\mathbb{Z}_7$.

Tabel 3.1. Pejumlahan untuk $\mathbb{Z}_7$.

+	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$
$\bar{0}$	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$
$\bar{1}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{0}$
$\bar{2}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{0}$	$\bar{1}$
$\bar{3}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{0}$	$\bar{1}$	$\bar{2}$
$\bar{4}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$
$\bar{5}$	$\bar{5}$	$\bar{6}$	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$
$\bar{6}$	$\bar{6}$	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$

Tabel 3.2. Perkalian untuk $\mathbb{Z}_7$.

.	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$
$\bar{0}$	$\bar{0}$	$\bar{0}$	$\bar{0}$	$\bar{0}$	$\bar{0}$	$\bar{0}$	$\bar{0}$
$\bar{1}$	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$
$\bar{2}$	$\bar{0}$	$\bar{2}$	$\bar{4}$	$\bar{6}$	$\bar{1}$	$\bar{3}$	$\bar{5}$
$\bar{3}$	$\bar{0}$	$\bar{3}$	$\bar{6}$	$\bar{2}$	$\bar{5}$	$\bar{1}$	$\bar{4}$
$\bar{4}$	$\bar{0}$	$\bar{4}$	$\bar{1}$	$\bar{5}$	$\bar{2}$	$\bar{6}$	$\bar{3}$
$\bar{5}$	$\bar{0}$	$\bar{5}$	$\bar{3}$	$\bar{1}$	$\bar{6}$	$\bar{4}$	$\bar{2}$
$\bar{6}$	$\bar{0}$	$\bar{6}$	$\bar{5}$	$\bar{4}$	$\bar{3}$	$\bar{2}$	$\bar{1}$

2.4. POLINOMIAL CHEBYSHEV ATAS LAPANGAN HINGGA $\mathbb{Z}_p$

Definisi dan sifat-sifat polinomial Chebyshev pada uraian subbab 2.2 juga berlaku untuk polinomial Chebyshev atas lapangan hingga $\mathbb{Z}_p$ (Hongzhou, Yun, & Dequan, 2006). Polinomial Chebyshev $T_n(x)$ atas lapangan hingga $\mathbb{Z}_p$ merupakan polinomial Chebyshev dengan variabel $x \in \mathbb{Z}_p$ dan koefisien-koefisien $T_n(x)$ juga $\in \mathbb{Z}_p$.

Contoh 2.11. Pandang polinomial Chebyshev $T_6(x) = 32x^6 - 48x^4 + 18x^2 - 1$. Polinomial chebyshev $T_6(x)$ atas lapangan hingga $\mathbb{Z}_{23}$ adalah polinomial Chebyshev dengan variabel $x \in \{\bar{0}, \bar{1}, \bar{2}, \dots, \overline{22}\}$ dan koefisien-koefisien $T_6(x)$ juga anggota dari $\{\bar{0}, \bar{1}, \bar{2}, \dots, \overline{22}\}$, yaitu

$$T_6(x) mod\ 23 = \left(32x^6 - 48x^4 + 18x^2 - 1\right) mod\ 23$$
$$= (32x^6) mod\ 23 - (48x^4)\ mod\ 23 + (18x^2)\ mod\ 23 - 1\ mod\ 23$$
$$= \left((9x^6) mod\ 23 - (2x^4) mod\ 23 + (18x^2)\ mod\ 23 - 1\ mod\ 23\right) mod\ 23.$$

BAB 3
KUNCI RAHASIA BERDASARKAN POLINOMIAL CHEBYSHEV PADA PROSES ENKRIPSI DAN DEKRIPSI

Baik pada enkripsi maupun dekripsi, kedua proses tersebut memerlukan bantuan suatu kunci. Dalam bab ini dibahas bagaimana membentuk suatu kunci berdasarkan polinomial Chebyshev pada algoritma Diffie-Hellman. Kemudian kunci tersebut digunakan dalam proses enkripsi dan dekripsi.

Pembahasan pada bab 3 dimulai dengan proses pembentukan kunci berdasarkan polinomial Chebyshev pada algoritma Diffie-Hellman, enkripsi dan dekripsi, algoritma-algoritma yang diperlukan dalam simulasi pembangkitan kunci, enkripsi dan dekripsi, kemudian diakhiri dengan hasil uji coba setelah algoritma-algoritma tersebut diimplemetasikan dalam bahasa pemrograman Matlab.

3.1. PEMBANGKITAN KUNCI BERDASARKAN POLINOMIAL CHEBYSHEV PADA ALGORITMA DIFFIE HELLMAN

Pada subbab 2.1 telah dibahas tentang pembangkitan kunci pada algoritma Diffie-Hellman berdasarkan monomial x^n. Pada bagian ini akan dibahas tentang pembangkitan kunci pada algoritma Diffie Hellman berdasarkan polinomial Chebyshev. Polinomial Chebyshev memenuhi relasi rekursif yang ditunjukkan oleh teorema 2.4. Sifat komutatif juga berlaku pada polinomial Chebyshev di bawah operasi komposisi fungsi yang dinyatakan oleh teorema 2.6.

Berikut ini akan ditunjukkan pembangkitan kunci pada algoritma Diffie-Hellman berdasarkan polinomial Chebyshev, langkah-langkah yang dilakukan adalah:

1. Alice menentukan bilangan prima p dan bilangan bulat x, dengan $1 < x < p$.
2. Alice memilih suatu bilangan bulat m dengan $1 < m < p$.
3. Alice menghitung $e = T_m(x) \bmod p$.
4. Alice mengirim p, x, dan e kepada Bob.
5. Bob memilih bilangan bulat n dengan $1 < n < p$.

6. Bob menghitung $b = T_n(x) \bmod p$.
7. Bob mengirim b kepada Alice.
8. Alice menghitung kunci rahasia $c = T_m(b) \bmod p$.
9. Bob menghitung kunci rahasia $d = T_n(e) \bmod p$ (Hongzhou, Yun, & Dequan, 2006).

Kunci rahasia yang mereka bangkitkan adalah sama, karena

$$\begin{aligned} c &= T_m(b) \bmod p, \quad b = T_n(x) \bmod p \\ &= T_m(T_n(x) \bmod p) \bmod p \\ &= T_m(T_n(x)) \bmod p \\ &= T_{mn}(x) \bmod p, \quad \text{berdasarkan teorema 2.6} \\ &= T_{nm}(x) \bmod p \\ &= T_n(T_m(x)) \bmod p \\ &= T_n(T_m(x) \bmod p) \bmod p \\ &= T_n(e) \bmod p \\ &= d \end{aligned}$$

Contoh 3.1. Di bawah ini akan ditunjukkan suatu contoh pembangkitan kunci berdasarkan polinomial Chebyshev pada algoritma Diffie Hellman, langkah-langkah yang dilakukan adalah:

1. Alice memilih bilangan prima $p = 23$ dan bilangan bulat $x = 7$.
2. Alice memilih bilangan bulat $m = 2$.
3. Alice menghitung $e = T_2(7) \bmod 23 = 97 \bmod 23 = 5$.
4. Alice mengirimkan $p = 23,\ x = 7,$ dan $e = 5$ kepada Bob.
5. Bob memilih $n = 3$.
6. Bob menghitung $b = T_3(7) \bmod 23 = 1351 \bmod 23 = 17$.
7. Bob mengirimkan $b = 17$ kepada Alice.
8. Alice menghitung $c = T_2(17) \bmod 23 = 577 \bmod 23 = 2$.
9. Bob menghitung $d = T_3(5) \bmod 23 = 485 \bmod 23 = 2$.

Baik Bob maupun Alice membangkitan kunci rahasia yang sama yaitu 2, yang juga dapat diperoleh dari $T_6(7) \bmod 23 = 3650401 \bmod 23 = 2$.

3.2. PROSES ENKRIPSI DAN DEKRIPSI

Jika Bob bermaksud mengirim pesan rahasia kepada Alice, berupa $plaintext\, M \in \{1, 2, \dots, p-1\}$, maka langkah-langkah yang dilakukan adalah:

- **Proses Enkripsi**
  1. Bob mengenkrispsi *plaintext* M menjadi *ciphertext* C dengan menggunakan persamaan
     $$C = M^d \bmod p.$$
  2. Bob mengirim *Ciphertext* C kepada Alice.
- **Proses Dekripsi**
  1. Alice menentukan nilai c^{-1} dengan cara mencari nilai yang memenuhi persamaan
     $$c\, c^{-1} \bmod p = 1.$$
  2. Setelah menerima *ciphertext* C, Alice mendekripsinya menjadi *plaintext* M dengan menggunakan persamaan
     $$M = C^{c^{-1}} \bmod p. \qquad (3.1.)$$

(Hongzhou, Yun, & Dequan, 2006).

Persaman (3.1.) dapat mengembalikan *Ciphertext* C menjadi *plaintext* M, karena

$$\begin{aligned} C^{c^{-1}} &= M^{c^{-1}\, d} \bmod p, \quad d = c \\ &= M^{c^{-1}\, c} \bmod p \\ &= M \bmod p \end{aligned}$$

Contoh 3.2. Berikut ini akan ditunjukkan suatu contoh enkripsi dan dekripsi berdasarkan pembangkitan kunci pada contoh 3.1. Pada contoh 3.1. telah diketahui bahwa nilai $c = d = 2$. Jika Bob bermaksud mengirim *plaintext* $M = 21$ kepada Alice, maka langkah-langkah yang dilakukan adalah:

- **Proses Enkripsi**
  1. Bob mengenkripsi $M = 21$ menjadi *ciphertext*
     $$C = 21.2 \bmod 23 = 42 \bmod 23 = 19.$$
  2. Bob mengirim *ciphertext* $C = 19$ kepada Alice.

- **Proses Dekripsi**

  1. Alice menentukan *invers* dari $c = 2$ terhadap operasi perkalian modulo p dengan cara mencari nilai yang memenuhi persamaan

     $2\,c^{-1}\,mod\,23 = 1.$

     Nilai yang memenuhi adalah $c^{-1} = 12$.

  2. Setelah menerima *ciphertext* $C = 19$, Alice mendekripsinya menjadi *plaintext*

     $M = 12\,.\,19\,mod\,23 \;=\; 228\,mod\,23 = 21.$

3.3. ALGORITMA-ALGORITMA YANG DIPERLUKAN DALAM SIMULASI PEMBENTUKAN KUNCI, ENKRIPSI, DAN DEKRIPSI

Pembangkitan kunci adalah hal yang pertama kali diperlukan agar dapat pengirim dan penerima dapat bertukar informasi rahasia. Setelah itu enkripsi dan dekripsi dapat dilakukan. Bagian ini mengulas tentang algoritma-algoritma yang digunakan dalam simulasi pembangkitan kunci, enkripsi, dan dekripsi.

Algoritma 3.1. Simulasi pembangkitan kunci.

Input: bilangan bulat positif p, x, m, dan n.

Output: bilangan bulat positif e, b, c, dan d.

Langkah-langkah:

1. mengentri nilai p
2. menyelidiki apakah p bilangan prima? Jika tidak maka kembali ke langkah 1, jika ya maka lanjut ke langkah 3
3. mengentri nilai x
4. menyelidiki apakah $1 < x < p$? Jika tidak maka kembali ke langkah 3, jika ya maka lanjut ke langkah 5
5. mengentri nilai m
6. menyelidiki apakah $1 < m < p$? Jika tidak maka kembali ke langkah 5, jika ya maka lanjut ke langkah 7
7. menentukan nilai $T_m(x)$
8. menentukan nilai $e = T_m(x)\ mod\, p$
9. mengentri nilai n

10. menyelidiki apakah $1 < n < p$? Jika tidak maka kembali ke langkah 9, jika ya maka lanjut ke langkah 11
11. menentukan nilai $T_n(x)$
12. menentukan nilai $b = T_n(x) \bmod p$
13. menentukan nilai $c = T_m(b) \bmod p$
14. menentukan nilai $d = T_n(e) \bmod p$
15. menampilkan nilai e, b, c, dan d.

Algoritma 3.2. Simulasi proses enkripsi/dekripsi.

Input: Pesan sebenarnya $\in \{1, 2, \ldots, p-1\}$.

Output: Pesan yang akan dikirim.

Langkah-langkah

1. memanggil nilai d dari algoritma pembangkitan kunci.
2. mengentri pesan sebenarnya.
3. menentukan pesan yang dikirim.

Algoritma 3.3. Algoritma Euclide yang diperluas (Menezes, Oorschoot, & Vanstone, 1997).

Input: $a, b \in \mathbb{Z}$, dengan a $\geq b$.

Output: $d = gcd\,(a, b)$ dan $x, y \in \mathbb{Z}$ yang memenuhi $ax + by = d$

Langkah-langkah:

1. Jika $b = 0$ maka *set* $d \leftarrow a,\ x \leftarrow 1,\ y \leftarrow 0$, *output* (d, x, y)
2. *Set* $x_2 \leftarrow 1,\ x_1 \leftarrow 0,\ y_2 \leftarrow 0,\ y_1 \leftarrow 1$
3. While $b > 0$:
   3.1. $q \leftarrow \left\lfloor \frac{a}{b} \right\rfloor,\ r \leftarrow a - qb,\ x \leftarrow x_2 - qx_1,\ y \leftarrow y_2 - qy_1$
   3.2. $a \leftarrow b,\ b \leftarrow r\ , x_2 \leftarrow x_1,\ x_1 \leftarrow x,\ y_2 \leftarrow y_1,\ y_1 \leftarrow y$
4. *Set* $d \leftarrow a, x \leftarrow x_2, y \leftarrow y_2$, *output* (d, x, y)

Algoritma 3.4. Menentukan *invers* perkalian modulo p (Menezes, Oorschoot, & Vanstone, 1997).

Input: $a \in \mathbb{Z}_p$.

Output: $a^{-1} \bmod p$.

Langkah-langkah:

1. tentukan $x, y \in \mathbb{Z}$ yang memenuhi $ax + py = d$, dengan $d = gcd(a, p)$ menggunakan algoritma 3.3.
2. jika $d > 1$ maka $a^{-1} \bmod p$ tidak ada, jika tidak maka *output* (x).

Algoritma 3.5. Menu utama

Input: bilangan 1, 2, 3, atau 4.

Ouput: program simulasi pembangkitan kunci dan program simulasi enkripsi/dekripsi.

Langkah-langkah:

1. mengentri salah satu dari angka 1, 2, 3, atau 4.
2. jika entri bukan angka 1, 2, 3, atau 4 maka kembali ke menu utama
3. jika entri adalah angka 1 maka memanggil program simulasi pembangkitan kunci
4. jika entri adalah angka 2 maka memanggil program simulasi enkripsi/dekripsi
5. jika entri adalah angka 3 maka memanggil program *inversemod* dan program simulasi dekripsi/dekripsi.
6. jika entri angka 4 maka keluar dari menu utama.

3.4. BEBERAPA HASIL UJI COBA SETELAH ALGORITMA-ALGORITMA DIIMPLEMETASIKAN

Di bawah ini adalah sejumlah hasil uji coba setelah algoritma-algoritma pada subbab 3.3 diimplemetasikan dengan menggunakan bahasa pemrograman Matlab.

Hasil Uji Coba 1:

```
--------------------------------
Menu Pembentukan Kunci
--------------------------------
Pilih bilangan prima p = 23
Masukkan nilai 1<x<p! 7
Masukkan nilai 1<m<p! 2
Masukkan nilai 1<n<p! 3
Hasil perhitungan:
 e=5
 b=17
 c=2
 d=2
Kembali ke Menu Utama (y/t)?
```

```
y
Menu Utama :
1. Pembentukan kunci
2. Enkripsi
3. Dekripsi
4. Keluar
Pilihan anda (1-4): 2
--------------------
 Menu Enkripsi
--------------------
Kunci sudah ada.
 p= 23
 x= 7
 c=d= 2
Masukkan pesan bilangan bulat positif<p : 21
Pesan yang dikirim:
    19
Kembali ke Menu Utama (y/t)?
y
Menu Utama :
1. Pembentukan kunci
2. Enkripsi
3. Dekripsi
4. Keluar
Pilihan anda (1-4): 3
-------------------
 Menu Dekripsi
-------------------
Kunci sudah ada.
 p= 23
 x= 7
 c=d= 2
Masukkan pesan bilangan bulat positif<p : 19
Pesan yang dikirim:
    21
```

Hasil Uji Coba 2:

```
---------------------------------
Menu Pembentukan Kunci
---------------------------------
Pilih bilangan prima p = 131
Masukkan nilai 1<x<p! 4
Masukkan nilai 1<m<p! 2
Masukkan nilai 1<n<p! 7
Hasil perhitungan:
 e=31
 b=8
```

```
 c=127
 d=127
Kembali ke Menu Utama (y/t)?
y
Menu Utama :
1. Pembentukan kunci
2. Enkripsi
3. Dekripsi
4. Keluar
Pilihan anda (1-4): 2
-------------------
 Menu Enkripsi
-------------------
Kunci sudah ada.
 p= 131
 x= 4
 c=d= 127
Masukkan pesan bilangan bulat positif<p : [43 56]
Pesan yang dikirim:
   90    38
Kembali ke Menu Utama (y/t)?
y
Menu Utama :
1. Pembentukan kunci
2. Enkripsi
3. Dekripsi
4. Keluar
Pilihan anda (1-4): 3
-------------------
 Menu Dekripsi
-------------------
Kunci sudah ada.
 p= 131
 x= 4
 c=d= 127
Masukkan pesan bilangan bulat positif<p : [90 38]
Pesan yang dikirim:
   43    56
```

Hasil Uji Coba 3:

```
---------------------------------
Menu Pembentukan Kunci
---------------------------------
Pilih bilangan prima p = 1223
Masukkan nilai 1<x<p! 8
Masukkan nilai 1<m<p! 5
Masukkan nilai 1<n<p! 3
Hasil perhitungan:
```

```
 e=428
 b=801
 c=726
 d=726
Kembali ke Menu Utama (y/t)?
y
Menu Utama :
1. Pembentukan kunci
2. Enkripsi
3. Dekripsi
4. Keluar
Pilihan anda (1-4): 2
-------------------
 Menu Enkripsi
-------------------
Kunci sudah ada.
 p= 1223
 x= 8
 c=d= 726
Masukkan pesan bilangan bulat positif<p : [12 345 1221]
Pesan yang dikirim:
   151   978   994
Kembali ke Menu Utama (y/t)?
y
Menu Utama :
1. Pembentukan kunci
2. Enkripsi
3. Dekripsi
4. Keluar
Pilihan anda (1-4): 3
--------------------
 Menu Dekripsi
--------------------
Kunci sudah ada.
 p= 1223
 x= 8
 c=d= 726
Masukkan pesan bilangan bulat positif<p : [151 978 994]
Pesan yang dikirim:
         12         345        1221
```

BAB 4
KESIMPULAN DAN SARAN

4.1. KESIMPULAN

Diawali dengan permasalahan apakah monomial x^n pada algoritma Diffie Hellman dapat diganti dengan polinomial Chebyshev. Kemudian bagaimanakah proses enkripsi dan dekripsi yang bersesuaian. Berdasarkan penelitian yang telah dilakukan diperoleh:

1. Monomial x^n pada algoritma pertukaran kunci Diffie Hellman dapat diganti dengan polinomial Chebyshev.
2. Proses enkripsi yang bersesuaian yaitu
   a. Pengirim pesan mengenkrispsi *plaintext* $M \in \{1, 2, \dots, p-1\}$ menjadi *ciphertext* C dengan menggunakan persamaan
      $$C = M\, d \bmod p$$
      dengan d adalah kunci rahasia dan p adalah bilangan prima.
   b. Pengirim pesan mengirim *Ciphertext* C kepada penerima pesan.
3. Proses dekripsi yang bersesuaian yaitu
   a. Penerima pesan menentukan nilai c^{-1} dengan cara mencari nilai yang memenuhi persamaan
      $$c\, c^{-1} \bmod p = 1$$
      dengan c adalah kunci rahasia, dan $c = d$.
   b. Setelah menerima *ciphertext* C, penerima pesan mendekripsinya menjadi *plaintext* M dengan menggunakan persamaan
      $$M = c^{-1}\, C \bmod p.$$

4.2. SARAN

Penelitian dalam tesis ini tidak mencakup aspek keamanan algoritma pada pembentukan kunci, proses enkripsi, dan dekripsi. Hal tersebut dapat menjadi topik penelitian selanjutnya.

DAFTAR REFERENSI

Alwi, H., et al. (2007). Kamus Besar Bahasa Indonesia Edisi Ketiga. Jakarta: Balai Pustaka.

Arifin, A. (2001). Aljabar Linier Edisi Kedua. Bandung: Penerbit ITB.

Clapham, C. (1996). *The Concise Oxford Dictionary of Mathematics Second Edition.* Oxford *University Press*.

Fee, G. J & Monagan, M. B. (2005). *Cryptography Using Chebyshev Polynomials*. Februari 2, 2011. http://www.cecm.sfu.ca/

Herstein, I. N. (1996). *Abstract Algebra*. New Jersey: Pentince-Hall, *Inc*.

Hongzhou N., Yun L., & Dequan H. (2006). *Public Key Encryption Algortihm Based on Chebyshev Polynomials over Finite Fields.* IEEE Xplore *Digital Library*.

Hungerford, T. W. (2003). *Graduate Text in Mathematics: Algebra.* Springer.

Juari, A. (2006). Aplikasi Teori Bilangan dalam Kriptografi Kunci Publik dan Algoritma Diffie Hellman. Februari 2, 2011. http://www.informatika.org/

Khuri, A. I. (2003). *Advanced Calculus with Application in Statistics. Second Edition.* Wiley *Series in Probability and Statistics.*

Koracev L. & Tasev Z. (2003). *Public key Encryption Algorithm Based on Chebyshev Maps*. IEEE Xplore *Digital Library*.

Mason, J. C & Handscomb, D.C. (2003). *Chebyshev Polynomials.* Chapman & Hall/CRC.

Menezes, A. J., van Oorschot P. C., & Vanstone S. A. (1997). *Handbook of Applied Cryptography*. New York: CRC *Press*.

Piper, F. & Murphy, S. (2002). *Cryptography: A very Short Introduction.* Oxford *University Press.*

Shoup, V. (2005). *A computational Introduction to Number Theory and Algebra.* Cambridge *University Press.*

Zimmermann, P. (2004). *An Introduction to Cryptography*. PGP *Corporation*.

Lampiran 1. Program Simulasi Pembangkitan Kunci

```
%fungsi pembentukan kunci berdasarkan polinomial Chebysev
%input : bilangan Prima p (kunci publik)
%        bilangan bulat 1<x<p (kunci publik)
%        bilangan bulat 1<m<p (kunci privat penerima pesan)
%        bilangan bulat 1<n<p (kunci privat pengirim pesan)
%
%output: sebuah file yang berisi kunci publik dan kunci rahasia

function keygen
clear all;
clc;
disp('----------------------');
disp('Menu Pembentukan Kunci');
disp('----------------------');
p=input('Pilih bilangan prima p = ');
while(~isprime(p))
    p=str2num(salah());
end
x=input('Masukkan nilai 1<x<p! ');
while((x<=1 || x>=p )&& mod(x,1)~=0)
    x=str2num(salah());
end

% kunci privat penerima pesan: m
m=input('Masukkan nilai 1<m<p! ');
while((m<=1 || m>=p) && mod(m,1)~=0)
    m=str2num(salah());
end
tm=cheby(m,x);
e=mod(tm,p);
% Kunci publik penerima pesan: p, x, dan e
% Nilai p, x dan e dikirim ke pengirim pesan

% kunci privat pengirim pesan: n
n=input('Masukkan nilai 1<n<p! ');
while((n<=1 || n>=p )&& mod(n,1)~=0)
    n=str2num(salah());
end

% kunci rahasia pengirim pesan=c
% kunci rahasia penerima pesan=d
% dan c=d
tn=cheby(n,x);
b=mod(tn,p);
tmb=cheby(m,b);
tne=cheby(n,e);
c=mod(tmb,p);
d=mod(tne,p);
```

```
fprintf('Hasil perhitungan: \n e=%d\n b=%d\n c=%d\n
d=%d\n',e,b,c,d);
if(e==0 || b==0 || c==0 || d==0)
   disp('Nilai e, b, c, atau d sebaiknya tidak ada yang 0.');
else
    save key.mat
end
end

% fungsi untuk menyatakan bahwa input salah
function h=salah()
disp('Input salah.');
h=input('Silakan ulangi! ','s');
end

% fungsi untuk manghasilkan output polinomial Chebyshev Tm(x).
function T=cheby(m,x)
% T=cosh(m*acosh(x));
if m>=2
    t0=1;
    t1=x;
    for i=2:m
        t=2*x*t1-t0;
        %update nilai
        t0=t1;
        t1=t;
    end
    T=t;
else
    disp('Nilai m harus >=2.');
    T=0;
end
end
```

Lampiran 2. Program Simulasi Proses Enkripsi/Dekripsi

```
%function enkripsi/dekripsi
%input : kunci rahasia
%        bilangan prima p
%        pesan bilangan bulat posisitif <p
%output: pesan yang dikirim

function [cipher]=enkrip(pesan,seckey,p)
l=length(pesan);
krip=pesan;
cipher=pesan;
salah=0;
for i=1:l
    if (pesan(i)>0 && pesan(i)<p)
        krip(i)=pesan(i)*seckey;
        cipher(i)=mod(krip(i),p);
    else
        salah=1;
    end
end
if salah==1
    disp('Pesan tidak sesuai.');
    cipher=0;
else
    disp('Pesan yang dikirim:')
    disp(cipher);
end
end
```

Lampiran 3. Program Algoritma Euclide yang Diperluas

```
%algoritma extended euclid
%input : bilangan bulat non negatif a dan b, a>=b
%output: d=gcd(a,b)
%        integer x,y memenuhi ax+by=d

function [d,x,y]=extendeuclid(a,b)
%step1 :
if(b==0)
    d=a;
    x=1;
    y=0;
else
    %step2
    x2=1;
    x1=0;
    y2=0;
    y1=1;
    %step3
    while(b>0)
        %step3.1
        q=floor(a/b);
        r=a-q*b;
        x=x2-q*x1;
        y=y2-q*y1;
        %step3.2
        a=b;
        b=r;
        x2=x1;
        x1=x;
        y2=y1;
        y1=y;
    end
    %step 4
    d=a;
    x=x2;
    y=y2;
end
```

Lampiran 4. Progam Menentukan *Invers* Perkalian Modulo p

```
%fungsi invers modulo
%input : integer a
%output: a^-1 mod n,dibuktikan ada

function [i]=inversemod(a,b)
[d,x,y]=extendeuclid(a,b);
if d>1
    i=0;
else
    i=x;
end
```

Lampiran 5. Program Menu Utama

```
%fungsi menu utama menampilkan program pembentukan kunci, enkripsi,
%dan dekripsi
%input : pilihan 1-4
%output: file key.mat sebagai kunci default setiap menjalankan
%fungsi
%        File ciper.mat berisi data yang sudah di enkripsi
%        file pesan.mat berisi pesan yang mau dikirimkan.

function utama
clear all;
clc;
disp('-------------------------------------------------');
disp(' Pembentukan Kunci Berdasarkan Polinomial Chebyshev ');
disp('        Beserta Proses Enkripsi dan Dekripsi       ');
disp('-------------------------------------------------');
ulg=1;
while(ulg==1)
    disp('Menu Utama : ');
    disp('1. Pembentukan kunci');
    disp('2. Enkripsi ');
    disp('3. Dekripsi ');
    disp('4. Keluar ');
    pil=input('Pilihan anda (1-4): ');
    switch (pil)
        case {1}
            keygen;
            ulg=ulang;
        case {2}
            if(~exist('key.mat'))
              disp('Kunci belum dibentuk.');
            else
                load key.mat;
                disp('---------------');
                disp(' Menu Enkripsi ');
                disp('---------------');
                disp('Kunci sudah ada.');
                fprintf(' p= %d\n x= %d\n c=d= %d\n',p,x,c);
                pesan=input('Masukkan pesan bilangan bulat positif<p
: ','s');
                chip=enkrip(str2num(pesan),d,p);
                chip=char(chip);
            end
            ulg=ulang;
        case {3}
            if(~exist('key.mat'))
                disp('Kunci belum dibentuk.');
            else
                load key.mat
                disp('---------------');
                disp(' Menu Dekripsi ')
                disp('---------------');
                disp('Kunci sudah ada.');
```

```
                fprintf(' p= %d\n x= %d\n c=d= %d\n',p,x,c);
                pesan=input('Masukkan pesan bilangan bulat positif<p
: ','s');
                k=inversemod(c,p);
                chip=enkrip(str2num(pesan),k,p);
                chip=char(chip);
            end
            ulg=ulang;
        case {4}
            disp('Terima kasih.');
            ulg=0;
        otherwise
            disp ('Pilihan tidak tersedia.');
            ulg=ulang;
    end
end
end

%fungsi kembali ke Menu Utama ya/tidak
%input : karakter
%output: 1: kembali ke Menu Utama
%        0: Keluar dari Menu Utama
%        2: pilihan salah
function y=ulang
disp('Kembali ke Menu Utama (y/t)? ');
p=input('','s');
p=lower(p);
if p=='y'
    y=1;
elseif p=='t'
    y=0;
else
    y=2;
end
end
```