PENERAPAN ALGORITMA DNA-*VIGENERE CIPHER* DENGAN KUNCI CITRA *GRAYSCALE* PADA DATA TEKS

SKRIPSI

Oleh

Danil Prastika Trimaratus Sholehah
NIM 131810101030

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2017

PENERAPAN ALGORITMA DNA-*VIGENERE CIPHER* DENGAN KUNCI CITRA *GRAYSCALE* PADA DATA TEKS

SKRIPSI

diajukan guna memenuhi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan Program Studi Matematika (S1)
dan mencapai gelar Sarjana Sains

Oleh

Danil Prastika Trimaratus Sholehah
NIM 131810101030

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2017

PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. ibu Maryamah dan ayah Imam Sujoko yang selalu mendampingi dan mendukung melalui doa, tenaga dan materi;
2. nenek, kakak dan adik yang tercinta;
3. keluarga besar yang tiada henti memberi doa dan dukungan;
4. sahabat angkatan 2013 (ATLAS) yang selalu memberikan semangat, dukungan, dan kenangan;
5. almamater tercinta jurusan Matematika FMIPA Universitas Jember.

MOTTO

"*Man Jadda Wajada – Siapa bersungguh-sunguh, pasti akan berhasil*".[1]

[1]Ahmad Fuadi dari novel Negeri 5 Menara

PERNYATAAN

Saya yang bertanda tangan dibawah ini:

nama: Danil Prastika Trimaratus Sholehah

nim : 131810101030

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul "Penerapan Algoritma DNA-*Vigenere Cipher* dengan Kunci Citra *Grayscale* pada Data Teks" adalah benar-benar hasil karya sendiri, kecuali kutipan yang telah disebutkan sumbernya, belum pernah diajukan di institusi manapun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, Juni 2017

Yang menyatakan,

Danil Prastika Trimaratus Sholehah

NIM 131810101030

SKRIPSI

PENERAPAN ALGORITMA DNA-*VIGENERE CIPHER* DENGAN KUNCI CITRA *GRAYSCALE* PADA DATA TEKS

Oleh

Danil Prastika Trimaratus Sholehah
NIM 131810101030

Pembimbing

Dosen Pembimbing Utama : Ahmad Kamsyakawuni, S.Si, M.Kom.

Dosen Pembimbing Anggota : Kusbudiono, S.Si, M.Si.

PENGESAHAN

Skripsi berjudul "Penerapan Algoritma DNA-*Vigenere Cipher* dengan Kunci Citra *Grayscale* pada Data Teks " telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Tim Penguji:

Ketua,	Anggota I,
Ahmad Kamsyakawuni, S.Si, M.Kom.	Kusbudiono, S.Si, M.Si.
NIP. 197211291998021001	NIP. 197704302005011001
Anggota II,	Anggota III,
Dr. Firdaus Ubaidillah, S.Si., M.Si.	Drs. Rusli Hidayat, M.Sc.
NIP. 197006061998031003	NIP. 196610121993031001

Mengesahkan
Dekan,

Drs. Sujito, Ph.D.
NIP. 196102041987111001

RINGKASAN

Penerapan Algoritma DNA-*Vigenere Cipher* dengan Kunci Citra *Grayscale* pada Data Teks; Danil Prastika Trimaratus Sholehah, 131810101030; 2017; 51 Halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Teknologi yang berkembang saat ini telah membawa informasi menjadi sangat penting dan sangat berpengaruh dalam sistem keamanan data. Sebagai manusia, tentunya memiliki informasi atau data-data yang bersifat rahasia dan hanya dapat diakses oleh pihak tertentu. Agar data-data tersebut tersimpan dengan aman, maka perlu adanya suatu perlindungan data. Salah satu ilmu untuk melindungi keamanan data adalah kriptografi.

Kriptografi merupakan suatu ilmu untuk melindungi data dengan teknik pengacakan atau pengubahan data dalam bentuk kode tertentu yang sulit dianalisa.Dalam kriptografi diperlukan suatu kunci untuk proses enkripsi maupun dekripsinya. Penelitian ini menggunakan suatu kunci berupa citra *grayscale* dengan kedalaman 8-bit. Pada tahun 2017, Saputra dkk. telah melakukan proses enkripsi dengan kunci citra *grayscale* menggunakan algoritma *Vigenere Cipher*. Namun, algortima *Vigenere Cipher* tidaklah aman karena menggunakan perulangan kunci apabila kuncinya pendek sehingga dapat dideteksi menggunakan frekuensi analisis. Untuk mengatasi kelemahan ini maka digunakan algoritma DNA-*Vigenere Cipher*.

Algoritma yang digunakan merupakan perkembangan baru dalam ilmu kriptografi yaitu kriptografi DNA. Metode ini menggunakan serangkaian basa nitrogen dari DNA yang merupakan suatu materi genetik dari mahluk hidup sebagai proses enkripsi dan dekripsinya. Rangkaian DNA berupa A (*adenin*), C (*cytosin*), G (*guanine*), T (*thymine*) adalah pasangan-pasangan basa nitrogen yang dapat diubah menjadi pasangan biner 0 dan 1. Dengan kode biner ini dapat dilakukan enkripsi selanjutnya. Pada proses enkripsi digunakan penggabungan algoritma *Vigenere Cipher* dengan kriptografi DNA. Penggabungan ini dihasilkan

tabel DNA-*Vigenere Cipher* yang lebih efisien dari tabel *Vigenere Cipher* pada umumnya. Selain itu, DNA-*Vigenere Cipher* memberikan hasil enkripsi dengan tingkat keamanan yang lebih baik (Najaftorkaman dan Kazazi, 2015).

Data yang digunakan dalam penelitian ini adalah data berupa teks sebagai *plaintext*. *Plaintext* akan dikonversi menjadi bilangan biner sesuai nilai desimal dalam kode *ASCII*, sedangkan pembangkitan kunci dilakukan dengan mengambil nilai *pixel* dari citra *grayscale* yang nantinya juga dikonversi menjadi bilangan biner. Setelah itu *plaintext* dan kunci yang berupa bilangan biner dikonversi ke dalam kode DNA dan dienkripsi menggunakan tabel DNA-*Vigenere Cipher*. Hasil dari operasi ini akan didapatkan suatu *ciphertext* yang tidak lagi mengandung informasi *plaintext* yang ada. Proses dekripsi juga mampu mengembalikan *ciphertext* menjadi *plaintext* tanpa adanya informasi yang hilang. Analisis keamanan dari metode ini menunjukkan bahwa metode ini memiliki tingkat keamanan yang baik dan aman dari serangan frekuensi analisis.

PRAKATA

Puji syukur kepada Allah SWT atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul "Penerapan Algoritma DNA-*Vigenere Cipher* dengan Kunci Citra *Grayscale* pada Data Teks". Tugas akhir ini disusun untuk memenuhi salah satu syarat pada program pendidikan strata satu (S1) Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Pada kesempatan ini penulis mengucapkan terima kasih atas bantuan dan bimbingan dalam penyusunan tugas akhir ini, terutama kepada yang terhormat:

1. Drs. Sujito, Ph.D., selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember;
2. Kusbudiono, S.Si., M.Si., selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember;
3. Ahmad Kamsyakawuni, S.Si., M.Kom selaku Dosen Pembimbing Utama dan Kusbudiono, S.Si., M.Si. selaku Dosen Pembimbing Anggota;
4. Dr. Firdaus Ubaidillah, S.Si., M.Si. selaku Dosen Penguji I dan Drs. Rusli Hidayat, M.Sc. selaku Dosen Penguji II;
5. Kedua orang tua, ibu Maryamah dan ayah Imam Sujoko serta nenek, kakak dan adik yang selalu memberikan dukungan dan doa;
6. Dosen dan Karyawan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember;
7. Teman-teman satu angkatan yang telah memberikan banyak kenangan dan dukungan.

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan penyusunan tugas akhir ini. Akhirnya penulis berharap, semoga tugas akhir ini dapat bermanfaat.

Jember, Juni 2017

Penulis

DAFTAR ISI

Halaman

DAFTAR GAMBAR

Halaman

DAFTAR TABEL

Halaman

DAFTAR LAMPIRAN

Halaman

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Perubahan waktu telah membawa informasi menjadi sangat penting terutama dalam perkembangan teknologi khususnya dalam keamanan sistem data. Teknologi berupa komputer saat ini sangat dibutuhkan manusia untuk menyimpan suatu data yang bersifat rahasia dan hanya dapat diakses oleh orang-orang tertentu. Untuk melindungi data-data yang tersimpan tersebut, perlu dilakukan suatu proses perlindungan agar data-data yang ada tidak disalahgunakan oleh pihak yang tidak berwenang.

Kriptografi adalah salah satu ilmu untuk melindungi data dengan teknik pengubahan maupun pengacakan data dalam bentuk kode tertentu sehingga data tersebut tidak terlihat seperti aslinya. Banyak metode dalam kriptografi untuk mengenkripsi data menjadi sebuah sandi yang sulit untuk dianalisa. Secara umum, data (*plaintext*) dienkripsi dengan suatu kunci berupa media yang sama dengan *plaintext* itu sendiri. Saputra dkk. (2017), telah melakukan enkripsi suatu *plaintext* dengan kunci berupa suatu citra *grayscale* 8-bit menggunakan algortima *Vigenere Cipher*. Kunci dibangkitkan dengan mengkonversi nilai *pixel* yang terdapat dalam citra dengan karakter pada tabel *ASCII*. Hasil konversi kemudian menjadi kunci untuk proses enkripsi menggunakan algoritma *Vigenere Cipher*. Tetapi, dalam algoritma *Vigenere Cipher* menggunakan perulangan kunci apabila kuncinya pendek, sehingga untuk memperoleh *plaintext* cukup mudah dilakukan dengan pendekatan frekuensi analisis.

Kriptografi DNA merupakan perkembangan baru dalam ilmu kriptografi. Perlindungan informasi ini dilakukan dengan mengubah *plaintext* menjadi serangkaian basa nitrogen dalam DNA (*deoxyribo nucleic acid*). Diketahui bahwa sebuah barisan DNA mengandung empat buah basa nitrogen yakni A (*adenine*), C (*cytosine*), G (*guanine*), T (*thymine*), dimana A dan T, C dan G adalah pasangan-pasangan basa yang dapat diubah menjadi pasangan biner 0 dan 1. Dengan kode biner ini dapat dilakukan enkripsi selanjutnya (Song dan Qiao, 2015).

Pada tahun 2015, Najaftorkaman dan Kazazi telah menggabungkan algoritma *Vigenere Cipher* dengan kriptografi DNA untuk mengenkripsi suatu data. Dengan penggabungan ini dihasilkan tabel DNA-*Vigenere Cipher* yang lebih efisien dari tabel *Vigenere Cipher* pada umumnya. Selain itu, DNA-*Vigenere Cipher* memberikan hasil enkripsi dengan tingkat keamanan yang lebih baik.

Pada penelitian ini, penulis ingin mengajukan suatu metode baru dalam mengenkripsi data. Untuk mengatasi kelemahan pada algoritma *Vigenere Cipher*, penulis akan menggunakan modifikasi *Vinegere Cipher* dengan kriptografi DNA, dimana *plaintext* akan dikonversi menjadi bilangan biner sesuai nilai desimal dalam karakter *ASCII* (*American Standard Code for Information Interchange*), sedangkan pembangkitan kunci dilakukan dengan mengambil setiap nilai *pixel* pada citra *grayscale* kemudian mengkoversi ke dalam biner. Setelah itu *plaintext* dan kunci yang berupa bilangan biner dikonversi ke dalam kode DNA dan dienkripsi berdasarkan tabel *DNA-Vinegere Cipher*. Dari penelitian ini, penulis berharap bahwa metode yang diajukan ini akan memberikan tingkat keamanan yang lebih tinggi dari penelitian-penelitian sebelumnya.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah tertera, maka rumusan masalah ditekankan pada:

a. Bagaimana mengenkripsi data dengan kunci citra *grayscale* pada algoritma DNA-*Vigenere Cipher*.
b. Bagaimana mendekripsi data yang telah dienkripsi.
c. Bagaimana analisis keamanan dari metode yang diajukan.

1.3 Batasan Masalah

Batasan masalah dari penelitian ini adalah *plaintext* yang digunakan bukan berupa kode ASCII *Control Character* (*Character code* 0-31).

1.4 Tujuan

Tujuan dari penelitian ini adalah:

a. Mengenkripsi data dengan kunci citra *grayscale* pada algoritma DNA-*Vigenere Cipher*.
b. Mendekripsi data yang telah dienkripsi.
c. Menganalisis keamanan metode yang diajukan.

1.5 Manfaat

Manfaat yang ingin diperoleh dengan adanya penelitian ini adalah:

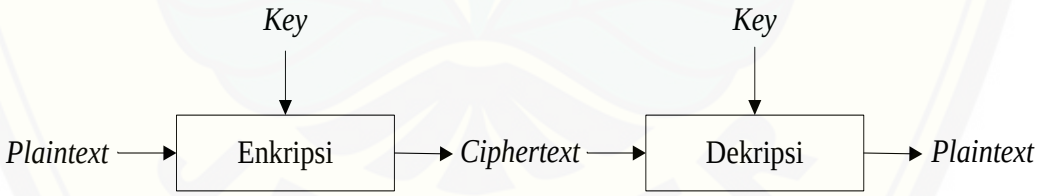
a. Mengetahui metode serta proses untuk mengenkripsi data dengan metode yang diajukan.
b. Mengetahui hasil dari proses enkripsi serta dekripsi dari metode yang diajukan.
c. Mampu menganalisis keamanan dari metode yang diajukan.

BAB 2. TINJAUAN PUSTAKA

2.1 Kriptografi

Kriptografi adalah suatu ilmu sekaligus seni untuk menjaga keamanan pesan atau *message*. Kriptografi berasal dari dua kata bahasa Yunani, yaitu *cryptós* (rahasia) dan *gráphein* (tulisan). Oleh sebab itu, dapat diartikan bahwa kriptografi merupakan ilmu yang mempelajari tentang tulisan rahasia atau pesan tersembunyi. Secara garis besar, ilmu kriptografi mempelajari teknik untuk menyembunyikan, melindungi dan mengamankan suatu informasi dengan cara membuat suatu bentuk baru yang susah dipahami maknanya (Munir, 2006).

Awal mula perkembangan ilmu kriptografi sebatas dipahami sebagai ilmu tentang penyandian/penyembunyian pesan. Namun seiring berkembangnya teknologi saat ini, selain untuk menjaga kerahasiaan (*confidentiality*) pesan, kriptografi juga digunakan untuk menangani masalah keamanan yang mencakup keabsahan pengirim (*user authentication*), keaslian pesan (*message aunthentication*) dan anti-penyangkalan (*nonrepudiation*). Gambar 2.1 merupakan proses ekripsi dan dekripsi dalam kriptografi.



Gambar 2.1 Proses Enkripsi dan Dekripsi

Istilah penting dalam kriptografi yang harus diketahui adalah *plaintext*, *ciphertext*, enkripsi, dekripsi, kunci (*key*), dan algoritma. *Plaintext* merupakan informasi awal atau pesan yang bisa dibaca dan dimengerti maknanya. *Ciphertext* merupakan informasi hasil pesan *plaintext* yang sudah disandikan. Enkripsi adalah teknik untuk menyandikan *plaintext*. Dekripsi adalah teknik untuk mengembalikan *ciphertext* menjadi *plaintext* kembali. Kunci (*key*) berfungsi untuk mengatur dan

menjalankan suatu algoritma. Sedangkan, algoritma adalah suatu metode untuk melalukan proses enkripsi dan dekripsi tersebut (Prayudi, 2005).

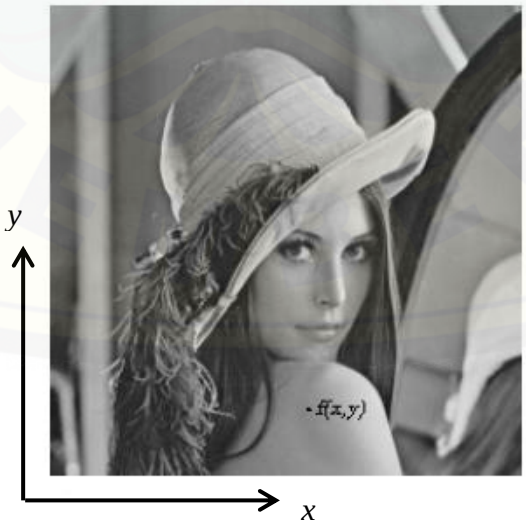
2.2 Citra

Citra (*image*) atau istilah lain untuk gambar adalah salah satu media yang memegang peranan sangat penting sebagai bentuk informasi visual. Citra dapat membentuk dua dimensi dan tiga dimensi untuk mempresentasikan bentuk suatu objek.

Menurut Murni (1992) bahwa citra sebagai keluaran dari suatu sistem perekam data dapat bersifat:

a. optik berupa foto,
b. analog berupa sinyal video seperti gambar pada monitor televisi,
c. digital yang dapat langsung disimpan pada suatu pita magnetik.

Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ dengan x dan y adalah suatu koordinat dan f dari (x,y) menyatakan intensitas atau derajat keabuan dari citra pada suatu titik. Derajat keabuan memiliki rentang nilai dari l_{min} sampai l_{max} atau $l_{min} \leq f \leq l_{max}$. Selang ($l_{min}$, l_{max}) disebut sebagai skala keabuan. Gambar 2.2 merupakan contoh representasi citra dalam suatu fungsi $f(x,y)$.



Gambar 2.2 Koordinat Titik pada Suatu Citra
(Sumber: Gonzalez, 1977)

Agar suatu citra dapat dilakukan proses komputasi pada komputer, citra harus didigitalisasi terlebih dahulu dimana proses digitalisasi merupakan representasi citra dari fungsi kontinu menjadi nilai-nilai diskrit. Hasil digitalisasi ini disebut sebagai citra digital. Citra digital dapat ditulis dalam bentuk matrik sebagai berikut:

$$f(x,y) = \begin{bmatrix} f(1,1) & \cdots & f(1,M) \\ \vdots & \ddots & \vdots \\ f(N,1) & \cdots & f(N,M) \end{bmatrix}$$

Indeks baris (*i*) dan indeks kolom (*j*) menyatakan suatu koordinat titik pada citra, dan *f*(*i*,*j*) merupakan intensitas (derajat keabuan) pada titik (*i*,*j*). Masing-masing elemen pada citra digital (elemen matriks) disebut dengan *pixel* atau *pel*. Jadi citra yang berukuran *N* x *M* mempunyai *NM* buah *pixel* (Dulimarta, 1997).

Citra *grayscale* merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap *pixel* nya, artinya nilai *Red* = *Green* = *Blue*. Nilai-nilai tersebut digunakan untuk menunjukkan intensitas warna. Citra yang ditampilkan terdiri atas warna abu-abu, bervariasi pada warna hitam sebagai bagian intensitas terlemah dan putih sebagai intensitas terkuat. Gambar 2.3 merupakan contoh citra *grayscale* berukuran 256x256 *pixel* dengan kedalaman 8-bit.

Gambar 2.3 Citra Rice
(Sumber: MATLAB *library*)

Citra *grayscale* berbeda dengan citra hitam-putih, dimana pada konteks komputer, citra hitam-putih hanya terdiri atas dua warna saja, yaitu hitam dan putih. Pada citra *grayscale*, warna bervariasi antara hitam dan putih menyebabkan

terdapat warna keabuan dengan berbagai tingkat dari hitam hingga mendekati putih. Umumnya citra *grayscale* direpresentasikan dalam 8-bit yang berarti terdapat 2^8 atau 256 derajat keabuan dengan rentang nilai 0-255, dimana 0 menunjukkan level intensitas paling gelap dan 255 menunjukkan intensitas paling terang.

2.3 Sistem Basis pada Bilangan

Sistem bilangan berdasarkan basisnya adalah bilangan desimal, bilangan biner, dan bilangan heksadesimal. Bilangan desimal merupakan bilangan yang memiliki basis 10, yaitu : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Bilangan biner merupakan bilangan yang memiliki basis 2, yaitu : 0 dan 1. Sedangkan bilangan heksadesimal memiliki basis 16, yaitu : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Suatu sistem basis bilangan dapat dikonversikan ke dalam sistem basis bilangan yang lainnya. Berikut merupakan beberapa langkah untuk melakukan konversi bilangan-bilangan tersebut.

a. Konversi Bilangan Desimal ke Bilangan Biner

   Salah satu cara dalam mengkonversi bilangan desimal menjadi bilangan biner adalah dengan cara membagi bilangan desimal dengan dua kemudian diambil sisa pembagiannya. Sisa-sisa pembagian membentuk jawaban, yaitu sisa yang pertama akan menjadi *least significant bit* (*LSB*) dan sisa yang terakhir menjadi *most significant bit* (*MSB*).

   Contoh :

   Bilangan desimal 45_{10}

   45 : 2 = 22 sisa 1 (*LSB*)

   22 : 2 = 11 sisa 0

   11 : 2 = 5 sisa 1

   5 : 2 = 2 sisa 1

   2 : 2 = 1 (*MSB*) sisa 0

   Bilangan biner ditulis dari bawah ke atas, maka bilangan biner dari 45_{10} adalah 101101_2

b. Konversi Bilangan Biner ke Bilangan Desimal

Sistem bilangan biner adalah susunan bilangan yang mempunyai basis 2 sebab sistem bilangan ini menggunakan dua nilai koefisien yang mungkin yaitu 0 dan 1. Konversi dilakukan dengan menggunakan persamaan (2.1).

$$D_r = \sum_i^{n-1}(d_i \times r^i) \quad (2.1)$$

dimana

r = basis bilangan biner yaitu 2

i = posisi nilai biner, dimulai dari 0

d = nilai biner

n = banyaknya digit biner

Contoh:

$101101_2 = 1x2^0 + 0x2^1 + 1x2^2 + 1x2^3 + 0x2^4 + 1x2^5$

$= 1 + 0 + 4 + 8 + 0 + 32$

$= 45$

c. Konversi Bilangan Biner ke Heksadesimal

Bilangan biner dibagi menjadi kelompok yang terdiri dari 4 digit biner. Setiap kelompok akan dikonversi menjadi 1 digit bilangan heksadesimal karena 4 digit bilangan biner bisa dikonversi menjadi 16 macam bilangan heksadesimal.

Contoh :

01001111_2

0100 = 4 dan 1111= F

Sehingga bilangan biner $01001111_2 = 4F_{16}$

d. Konversi Heksadesimal ke Bilangan Biner

Setiap digit heksadesimal langsung dikonversi menjadi bilangan biner lalu hasilnya dipadukan.

Contoh :

$F5_{16}$

F = 15 = 1111 dan 5 = 0101

Sehingga bilangan heksadesimal $F5_{16} = 11110101_2$

2.4 *ASCII* (*American Standard Code for Information Interchange*)

ASCII (*American Standard Code for Information Interchange*) merupakan suatu standar internasional dalam kode huruf dan simbol yang digunakan oleh komputer dan alat komunikasi lainnya untuk menunjukkan suatu teks. Kode *ASCII* memiliki komposisi sebanyak 8-bit atau 256 karakter, dimulai dari 0 hingga 255 sistem bilangan dan dibagi menjadi beberapa bagian, seperti *ASCII Control Characters*, *ASCII Printable Character*, dan *The Extended ASCII Codes*. Kode *ASCII* dapat dilihat pada Lampiran A.

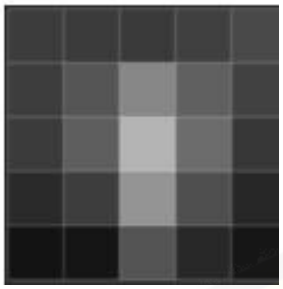
Kode *ASCII* merepresentasikan kode-kode untuk:

a. Angka (0,1,2,3,4,5,6,7,8,9)
b. Huruf (a – z, A – Z)
c. Simbol (&, ^, %, $, #, ...)
d. Tombol (Enter, Esc, Backspace, Space, Tab, Shift, Ctrl)
e. Karakter Grafis (kode ASCII standar nomor 128 s/d 255)
f. Kode komunikasi (ETX, STX, ENQ, ACK, ...)

2.5 *Vigenere Cipher* dengan Kunci Citra *Grasyscale*

Vigenere Cipher merupakan algoritma kriptografi klasik yang dikembangkan dari algoritma *Caesar Cipher* sehingga tergolong dalam metode substitusi abjad-majemuk. Dasar dari algoritma *Vigenere Cipher* menggunakan 26 karakter huruf (A-Z) sebagai media enkripsi dan dekripsinya. Proses enkripsi dan dekripsinya dilakukan dengan menggunakan tabel *Vigenere* dan membutuhkan kunci (*key*). Setiap baris di dalam tabel menyatakan huruf-huruf *ciphertext* dan setiap huruf *ciphertext* yang sama belum tentu berasal dari *plaintext* yang sama pula sehingga algoritma ini tergolong *cipher* abjad-majemuk.

Saputra dkk. (2017) telah melakukan proses enkripsi menggunakan algoritma *Vigenere Cipher*, tetapi kunci yang digunakan berupa citra *grayscale* dengan kedalaman 8-bit seperti pada Gambar 2.4. Setiap nilai *pixel* dari citra ini digunakan sebagai kunci untuk proses enkripsi dan dekripsi sesuai algoritma *Vigenere Cipher*. Berikut adalah proses enkripsi dan dekripsi menggunakan kunci citra *grayscale*.



48	41	39	42	48
46	58	134	74	44
46	37	142	88	45
26	42	152	72	20
15	20	44	26	14

Gambar 2.4 Kunci dengan Citra *Grayscale* 5x5 *pixel* dan Kedalaman 8-bit

(Sumber: Saputra dkk., 2017)

a. Proses Enkripsi

Plaintext yang digunakan adalah NEVER UNDERESTIMATE YOURSELF. Sebelum dienkripsi menggunakan kunci, *plaintext* harus dikonversi menjadi bilangan desimal sesuai karakter *ASCII*.

Persamaan enkripsi *Vigenere Cipher*:

$C_i = (P_i + K_i) \bmod 256$

dimana C_i = *Ciphertext*

P_i = *Plaintext*

K_i = *Key* atau kunci

$C_0 = (78 + 48) \bmod 256 = 126$, membentuk karakter ~

$C_1 = (69 + 41) \bmod 256 = 110$, membentuk karakter n

$C_2 = (86 + 39) \bmod 256 = 125$, membentuk karakter }

Untuk C_n dilakukan dengan cara yang sama sehingga *ciphertext* yang dihasilkan adalah ~n}oéNÅÈÄqÇjß¯vgmýÍ4hcùlauum

b. Proses Dekripsi

Persamaan dekripsi *Vigenere Cipher* :

$P_i = (C_i - K_i) \bmod 256$

$P_0 = (126 - 48) \bmod 256 = 78$, membentuk karakter N

$P_1 = (110 - 41) \bmod 256 = 69$, membentuk karakter E

$P_2 = (125 - 39) \bmod 256 = 86$, membentuk karakter V

Untuk P_n dilakukan dengan cara yang sama sehingga *plaintext* yang dihasilkan adalah NEVER UNDERESTIMATE YOURSELF

2.6 Kriptografi DNA

Kriptografi DNA merupakan ilmu yang baru dalam perlindungan informasi dimana ilmu ini didasarkan pada struktur dari DNA. Pada DNA terdapat 4 buah basa nitrogenyakni A, C, T, dan G dimana A & T serta C & G adalah pasangan komplemen. Dasar ini yang digunakan untuk melakukan proses perhitungan komputasinya.

Dalam sistem bilangan biner, 0 dan 1 adalah komplemen, yaitu 00 dan 11, 10 dan 01. Jika bilangan biner dikodekan menjadi pasangan komplemen pada DNA, maka terdapat 8 macam kombinasi pengkodean yang memenuhi aturan pasangan basa komplemen seperti yang terdapat pada Tabel 2.1. Untuk sebuah citra *grayscale* dengan kedalaman 8 bit, dapat dikodekan menjadi pasangan basa komplemen dengan panjang 4 (Song dan Qiao, 2015).

Tabel 2.1 Delapan Macam Bentuk Pengkodean DNA
(Sumber: Song dan Qiao, 2015)

	1	2	3	4	5	6	7	8
A	00	00	01	01	10	10	11	11
T	11	11	10	10	01	01	00	00
G	01	10	00	11	00	11	01	10
C	10	01	11	00	11	00	10	01

2.7 DNA-*Vigenere Cipher*

DNA-*Vigenere Cipher* merupakan modifikasi *Vigenere Cipher* dengan kriptografi DNA. Proses enkripsi dan dekripsinya berdasar pada *Vigenere Cipher*, hanya saja *plaintext*, *ciphertext*, serta *key* nya harus dikodekan menjadi basa nitrogen pada DNA terlebih dahulu dengan menggunakan Tabel 2.1. Tabel 2.2 menyatakan tabel *Vigenere* yang telah dimodifikasi dengan kriptografi DNA.

Berikut adalah contoh enkripsi dan dekripsi menggunakan DNA-*Vigenere Cipher*:

a. Contoh Enkripsi

Plaintext : CAGT

Key : GCGA

Ciphertext : TGCG

b. Contoh Dekripsi

Ciphertext : TGCG

Key : GCGA

Plaintext : CAGT

Tabel 2.2 Tabel DNA-*Vigenere Cipher*
(Sumber: Najaftorkaman dan Kazazi, 2015)

		Plaintext			
		A	T	C	G
Key	A	A	T	C	G
	T	T	C	G	A
	C	C	G	A	T
	G	G	A	T	C

2.8 Analisis Keamanan

Dalam proses perlindungan data terdapat beberapa metode yang dapat digunakan untuk menganalisis keamanan dari suatu algoritma yang digunakan. Masalah yang terdapat pada enkripsi berbasis *Vigenere Cipher* ialah perulangan dari kuncinya apabila kuncinya pendek. Jika panjang kunci dari *Vigenere Cipher* diketahui, maka seluruh *ciphertext* akan dapat dipecahkan karena panjang dari setiap bagian diketahui dan setelah itu setiap bagian akan digunakan untuk memperoleh *plaintext* dengan menggunakan pendekatan frekuensi analisis.

Kriptanalisis sendiri dapat dilakukan dengan bermacam-macam Bahasa di seluruh dunia, namun hanya beberapa saja yang baru dapat dipecahkan dan dianalisa. Selain itu pada kriptanalisis pun tidak sepenuhnya sempurna dalam memecahkan suatu *ciphertext*, maka dari itu diambil nilai kemungkinan atau *probability* yang sering disebut dengan *index coincidence. Index Coincidence* (IC) ini merupakan hubungan antara satu huruf dengan huruf lain yang dapat menghasilkan kemungkinan sebuah kunci. Rumus untuk mencari *index coincidence* adalah seperti pada persamaan (2.3). Nilai indeks pada tiap bahasa pun berbeda-beda nilainya, seperti pada Tabel 2.3 (Bawono, 2015).

Tabel 2.3 *Index of Coincidence*

Bahasa	*Index of Coincidence*
Inggris	0,067
Perancis	0,078
Jerman	0,076
Italia	0,074
Jepang	0,082
Rusia	0,053
Teks Acak	0,038

Tes Friedman merupakan salah satu metode yang digunakan untuk menentukan panjang kunci. Untuk mencari panjang kunci ini dilakukan dengan persamaan (2.2).

$$Key\ length = \left|\frac{K_p - K_r}{K_0 - K_r}\right| \tag{2.2}$$

dimana

K_p = Peluang dua buah *ciphertext* yang dipilih adalah sama. Dalam bahasa Inggris, nilainya adalah 0,067

K_r = Peluang dari terambilnya dua buah huruf yang sama dari alfabet. Di dalam kriptografi DNA, terdapat empat huruf yakni A, C, T, dan G. Sehingga nilainya adalah 0,25

K_0 = *Index coincidence* yang dirumuskan dengan persamaan (2.3)

$$K_0 = \frac{\sum_{i=1}^{c} f_i(f_i - 1)}{N(N-1)} \tag{2.3}$$

dimana *c* adalah banyaknya alfabet, *N* adalah panjang *ciphertext*, dan f_i adalah frekuensi kemunculan alfabet (Bevi dkk., 2016).

BAB 3. METODE PENELITIAN

3.1 Data Penelitian

Data yang digunakan dalam penelitian ini adalah pesan teks yang berlaku sebagai *plaintext* dan citra *grayscale* dengan kedalaman 8-bit sebagai kunci. Gambar 3.1 merupakan citra *grayscale* berdimensi 256x256 *pixel* dengan kedalaman 8-bit.

Gambar 3.1 Citra Cameraman
(Sumber: MATLAB *library*)

3.2 Langkah-langkah Penelitian

Secara sistematis, langkah-langkah penelitian yang dilakukan adalah sebagai berikut.

a. Studi Literatur

   Pada tahap ini dilakukan pemahaman mengenai teori-teori terkait dengan penelitian yang dilakukan. Teori-teori tersebut adalah definisi citra *grayscale*, algoritma *Vigenere Cipher*, dan kriptografi DNA.

b. Perancangan Program

   Pada tahap ini dilakukan perancangan desain GUI (*Guide User Interface*) dengan menggunakan *software* Matlab R2009a seperti tata-letak, tombol-

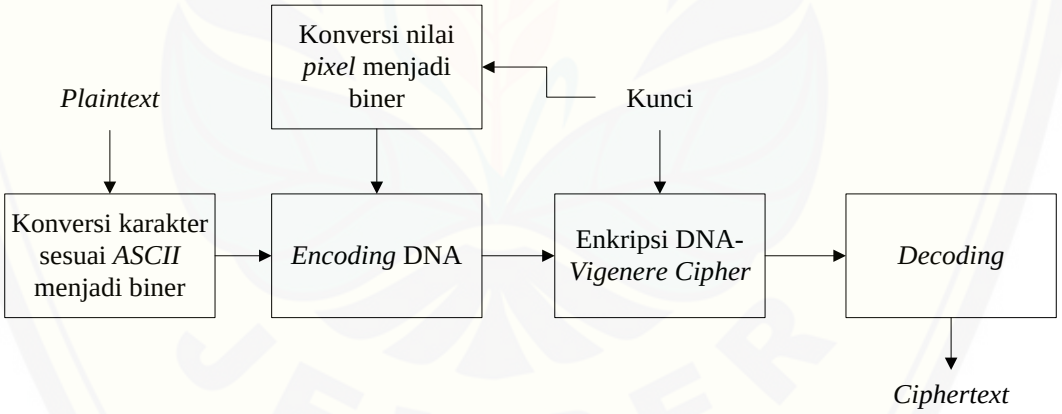
tombol serta pengaturan warna dan latar belakang agar tampilan menjadi menarik.

c. Pembuatan Program

Pembuatan program dilakukan berdasarkan konsep matriks sebagai pembangkit citra pada *software* Matlab R2009a, melakukan proses enkripsi dan dekripsi dengan algortima DNA-*Vigenere Cipher*. Berikut uraian proses enkripsi dan dekripsi.

1) Proses enkripsi

Pembentukan *plaintext* menjadi *ciphertext* dilakukan dengan mengikuti aturan dalam algoritma yang digunakan. Oleh karenanya, dalam penelitian ini dibentuk sebuah bagan proses enkripsi untuk mempermudah dan memperjelas langkah-langkah proses enkripsinya. Langkah-langkah berikut ini akan diterapkan pada seluruh data teks dan citra *grayscale* yang menjadi kunci. Proses enkripsi dilakukan seperti pada Gambar 3.2.



Gambar 3.2 Bagan Proses Enkripsi

a) Konversi karakter *plaintext* menjadi bilangan biner

Setiap karakter atau simbol dalam *plaintext* dikonversi menjadi bilangan desimal sesuai kode *ASCII* pada Lampiran A kemudian dikonversi menjadi bilangan biner.

b) Konversi nilai *pixel* menjadi biner

Pembangkitan kunci dilakukan dengan mengambil setiap nilai *pixel* dari citra *grayscale*. Nilai *pixel* yang berupa bilangan desimal ini kemudian dikonversi menjadi bilangan biner.

c) *Encoding* DNA

Nilai *plaintext* dan kunci yang telah dikonversi masih berupa bilangan biner, untuk itu akan dikodekan menjadi basa nitrogen DNA dengan menggunakan Tabel 2.1. *Encoding* dilakukan sesuai kode yang ditentukan.

d) Enkripsi menggunakan DNA-*Vigenere Cipher*

Plaintext dan kunci yang telah dikonversi menjadi basa nitrogen DNA akan dioperasikan berdasarkan Tabel 2.2. Hasil dari operasi ini akan menghasilkan *ciphertext* berupa serangkaian basa nitrogen DNA.

e) *Decoding*

Serangkaian basa nitrogen DNA hasil enkripsi dikonversi kembali menjadi bilangan biner berdasarkan Tabel 2.1. *Decoding* dilakukan sesuai kode pada proses *encoding*. Kemudian bilangan biner dari hasil *decoding* dikonversi menjadi bilangan heksadesimal. Hasil ini kemudian menjadi *ciphertext*.

2) Proses dekripsi

Seperti pada proses enkripsi, proses dekripsi juga dibentuk sebuah bagan untuk mempermudah dalam prosesnya. Proses dekripsi dilakukan seperti pada Gambar 3.3.

a) Konversi karakter *ciphertext* menjadi bilangan biner

Setiap karakter dalam *ciphertext* dikonversi menjadi bilangan biner.

b) Konversi nilai *pixel* menjadi biner

Pembangkitan kunci dilakukan dengan mengambil setiap nilai pixel dari citra *grayscale*. Nilai *pixel* diambil sesuai panjang dari *ciphertext*. Nilai *pixel* yang berupa bilangan desimal ini kemudian dikonversi menjadi bilangan biner.

c) *Encoding* DNA

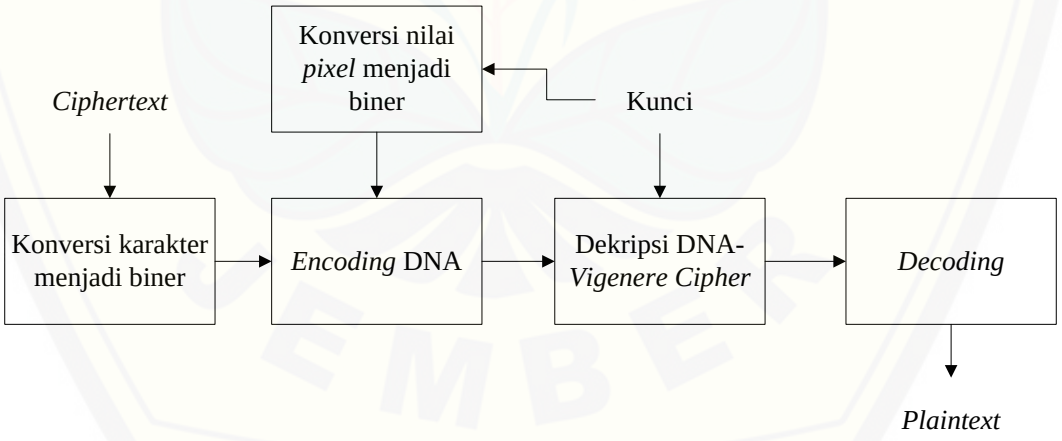
Nilai *ciphertext* dan kunci yang telah dikonversi masih berupa bilangan biner. Untuk itu akan dikodekan menjadi basa nitrogen DNA dengan menggunakan Tabel 2.1 sesuai dengan kode yang digunakan dalam proses enkripsi.

d) Dekripsi menggunakan DNA-*Vigenere Cipher*

Ciphertext dan kunci yang telah dikonversi menjadi basa nitrogen DNA akan dioperasikan berdasarkan Tabel 2.2. Hasil dari operasi ini akan menghasilkan serangkaian basa nitrogen DNA.

e) *Decoding*

Serangkaian basa nitrogen DNA hasil enkripsi dikonversi kembali menjadi bilangan biner berdasarkan Tabel 2.1, kemudian dikonversi menjadi bilangan desimal dengan persamaan (2.1) sehingga membentuk simbol dalam kode *ASCII*. Hasil *decoding* ini kemudian menjadi *plaintext*.



Gambar 3.3 Bagan Proses Dekripsi

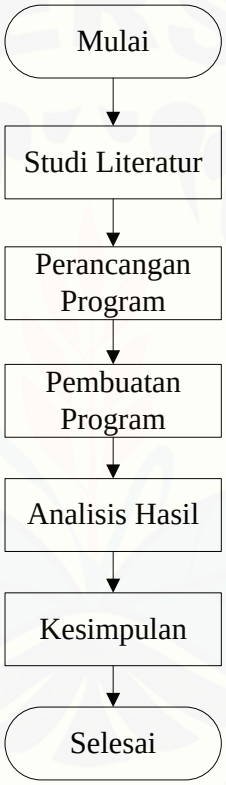
d. Analisis Hasil

Menguji jalannya program yang telah dibuat untuk menentukan apakah setiap proses telah berjalan dengan baik sesuai dengan hasil yang diinginkan atau tidak. Kemudian diuji keamanannya dengan analisis panjang kunci.

e. Kesimpulan

Mengambil kesimpulan dari penelitian yang dilakukan, yaitu menganalisis proses enkripsi dalam mengubah *plaintext* menjadi *ciphertext* dan sebaliknya dengan kunci citra *grayscale* menggunakan metode yang diajukan, serta analisis keamanannya.

Flowchart dari langkah-langkah penelitian yang dilakukan dapat diamati pada Gambar 3.4.



Gambar 3.4 *Flowchart* Penelitian

BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian yang dilakukan, maka dapat diambil beberapa kesimpulan sebagai berikut.

a. Proses enkripsi dengan algoritma DNA-*Vigenere Cipher* menggunakan kunci citra *grayscale* dapat memberikan keamanan yang baik karena proses ini memperkecil analisa kriptanalis serta mempermudah penerima pesan untuk mengingat kunci, karena kunci yang digunakan berupa citra.
b. Proses dekripsi menggunakan algoritma DNA-*Vigenere Cipher* dengan kunci citra *grayscale* mampu mengembalikan *ciphertext* menjadi *plaintext* tanpa adanya informasi yang hilang.
c. Metode yang diajukan memiliki tingkat keamanan yang baik karena menggunakan kunci citra *grayscale* serta kunci yang digunakan tidak dapat dipecahkan.

5.2 Saran

Saran yang diberikan untuk penelitian selanjutnya adalah:

a. Citra yang digunakan sebagai kunci tidak terbatas pada citra *grayscale*, tetapi bisa juga digunakan dengan citra RGB atau citra warna.
b. Menerapkan kriptografi DNA pada algoritma modern seperti DES, AES, dsb.

DAFTAR PUSTAKA

Bevi, A. R., K. Patel, dan S. Malarvizhi. 2016. Performance Analysis of Hardware Implemented DNA Algorithm for Security Applications. *ARPN Journal Of Engineering and Applied Science* 11(13) : 8049-8056.

Bawono, H. R. C. 2015. Kriptanalis Pada Algoritma Cipher Vigenere. *Skripsi*. Yogyakarta: Fakultas Sains dan Teknologi Universitas Sanata Dharma.

Dulimarta, H.S. 1997. *Diktat Kuliah Pengolahan Citra*. Bandung : Jurusan Teknik Informatika Institut Teknologi Bandung.

Gonzalez, R.C. 1977. *Digital Image Processing.* USA : Addison-Wesley Publishing.

Munir, R. 2006. *Diktat Kuliah IF5054 Kriptografi.* Jakarta : Program Studi Teknik Informatika, Sekolah Teknik Elektro dan Informatika.

Murni, A. 1992. *Pengantar Pengolahan Citra*. Jakarta : PT Elex Media Komputindo.

Najaftorkaman M., dan N. S. Kazazi. 2015. A Method to Encrypt Information with DNA-Based Cryptography. *International Journal of Cyber-Security and Digital Forensics* (*IJCSDF*) 4(3): 417-426.

Prayudi. 2005. *Studi Analisis Algoritma Rivest Code 6* (*RC6*) *Dalam Enkripsi/Dekripsi Data*. Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNATI 2005), Yogyakarta.

Saputra, I., Mesran, N. A. Hasibuan, dan R. Rahim. 2017. Vigenere Cipher Algorithm with Grayscale Image Key Generator for Secure Text File. *International Journal of Engineering Research & Technology* (*IJERT*) 6 : 266-269.

Song, C., dan Y. Qiao. 2015. A Novel Image Encryption Algorithm Based on DNA Encoding and Spatiotemporal Chaos. *Entropy* 17 : 6954-6968.

LAMPIRAN

LAMPIRAN A. Kode *ASCII*

Kode *ASCII Control Character* (*Character code* 0-31)

Dec	*Hex*	*Symbol*	*Dec*	*Hex*	*Symbol*
0	0	NUL	16	10	DLE
1	1	SOH	17	11	DC1
2	2	STX	18	12	DC2
3	3	ETX	19	13	DC3
4	4	EOT	20	14	DC4
5	5	ENQ	21	15	NAK
6	6	ACK	22	16	SYN
7	7	BEL	23	17	ETB
8	8	BS	24	18	CAN
9	9	HT	25	19	EM
10	A	LF	26	1A	SUB
11	B	VT	27	1B	ESC
12	C	FF	28	1C	FS
13	D	CR	29	1D	GS
14	E	SO	30	1E	RS
15	F	SI	31	1F	US

Kode *ASCII Printable Character* (*Character code* 32-127)

Dec	*Hex*	*Symbol*	*Dec*	*Hex*	*Symbol*
32	20	Spasi	80	50	P
33	21	!	81	51	Q
34	22	”	82	52	R
35	23	#	83	53	S
36	24	$	84	54	T
37	25	%	85	55	U
38	26	&	86	56	V
39	27	’	87	57	W
40	28	(	88	58	X
41	29	)	89	59	Y
42	2A	*	90	5A	Z
43	2B	+	91	5B	[
44	2C	,	92	5C	\
45	2D	-	93	5D	]
46	2E	.	94	5E	^
47	2F	/	95	5F	_
48	30	0	96	60	~
49	31	1	97	61	a

Dec	*Hex*	*Symbol*	*Dec*	*Hex*	*Symbol*
50	32	2	98	62	b
51	33	3	99	63	c
52	34	4	100	64	d
53	35	5	101	65	e
54	36	6	102	66	f
55	37	7	103	67	g
56	38	8	104	68	h
57	39	9	105	69	i
58	3A	:	106	6A	j
59	3B	;	107	6B	k
60	3C	<	108	6C	l
61	3D	=	109	6D	m
62	3E	>	110	6E	n
63	3F	?	111	6F	o
64	40	@	112	70	p
65	41	A	113	71	q
66	42	B	114	72	r
67	43	C	115	73	s
68	44	D	116	74	t
69	45	E	117	75	u
70	46	F	118	76	v
71	47	G	119	77	w
72	48	H	120	78	x
73	49	I	121	79	y
74	4A	J	122	7A	z
75	4B	K	123	7B	{
76	4C	L	124	7C	|
77	4D	M	125	7D	}
78	4E	N	126	7E	~
79	4F	O	127	7F	Delete

The Extended ASCII Codes (*Character code* 128-255)

Dec	*Hex*	*Symbol*	*Dec*	*Hex*	*Symbol*
128	80	€	192	C0	À
129	81		193	C1	Á
130	82	‚	194	C2	Â
131	83	*ƒ*	195	C3	Ã
132	84	„	196	C4	Ä
133	85	…	197	C5	Å
134	86	†	198	C6	Æ
135	87	‡	199	C7	Ç
136	88	ˆ	200	C8	È
137	89	‰	201	C9	É
138	8A	Š	202	CA	Ê

Dec	*Hex*	*Symbol*	*Dec*	*Hex*	*Symbol*
139	8B	‹	203	CB	Ë
140	8C	Œ	204	CC	Ì
141	8D		205	CD	Í
142	8E	Ž	206	CE	Î
143	8F		207	CF	Ï
144	90		208	D0	Đ
145	91	‘	209	D1	Ñ
146	92	’	210	D2	Ò
147	93	“	211	D3	Ó
148	94	”	212	D4	Ô
149	95	•	213	D5	Õ
150	96	–	214	D6	Ö
151	97	—	215	D7	×
152	98	˜	216	D8	Ø
153	99	™	217	D9	Ù
154	9A	š	218	DA	Ú
155	9B	›	219	DB	Û
156	9C	œ	220	DC	Ü
157	9D		221	DD	Ý
158	9E	ž	222	DE	Þ
159	9F	Ÿ	223	DF	ß
160	A0		224	E0	à
161	A1	¡	225	E1	á
162	A2	¢	226	E2	â
163	A3	£	227	E3	ã
164	A4	¤	228	E4	ä
165	A5	¥	229	E5	å
166	A6	¦	230	E6	æ
167	A7	§	231	E7	ç
168	A8	¨	232	E8	È
169	A9	©	233	E9	É
170	AA	ª	234	EA	Ê
171	AB	«	235	EB	Ë
172	AC	¬	236	EC	Ì
173	AD		237	ED	Í
174	AE	®	238	EE	Î
175	AF	¯	239	EF	Ï
176	B0	°	240	F0	Đ
177	B1	±	241	F1	Ñ
178	B2	²	242	F2	Ò
179	B3	³	243	F3	Ó
180	B4	´	244	F4	Ô
181	B5	µ	245	F5	Õ
182	B6	¶	246	F6	Ö
183	B7	·	247	F7	÷

Dec	*Hex*	*Symbol*	*Dec*	*Hex*	*Symbol*
184	B8	¸	248	F8	Ø
185	B9	¹	249	F9	Ù
186	BA	º	250	FA	Ú
187	BB	»	251	FB	Û
188	BC	¼	252	FC	ü
189	BD	½	253	FD	ý
190	BE	¾	254	FE	þ
191	BF	¿	255	FF	ÿ

LAMPIRAN B. Nilai *Pixel* dari Citra Cameraman pada Gambar 3.1

156	159	158	155	158	156	159	158	...	152
160	154	157	158	157	159	158	158	...	153
156	159	158	155	158	156	159	158	...	152
160	154	157	158	157	159	158	158	...	153
156	153	155	159	159	155	156	155	...	151
155	155	155	157	156	159	152	158	...	153
156	153	157	156	153	155	154	155	...	149
159	159	156	158	156	159	157	161	...	147
158	155	158	154	156	160	162	155	...	147
155	154	157	158	160	160	159	160	⋯	153
154	157	157	157	156	155	159	154	⋯	155
152	150	155	154	152	156	157	156	⋯	148
157	153	156	155	157	160	160	157	⋯	151
151	154	157	156	156	158	158	156	⋯	148
156	157	157	160	159	159	156	158	⋯	150
157	158	159	157	157	154	153	158	⋯	156
154	154	156	157	158	159	157	160	⋯	152
151	153	157	152	156	156	155	156	⋯	152
153	155	154	153	156	155	153	155	⋯	155
152	154	152	156	159	154	156	155	⋯	151
154	157	155	156	157	154	158	158	⋯	155
155	153	155	155	159	160	159	161	⋯	150
151	151	153	155	153	156	155	155	⋯	156
150	151	155	154	155	154	156	152	⋯	157
153	154	151	155	154	153	155	157	⋯	153
154	154	155	156	155	156	155	156	⋯	155
162	157	155	154	156	155	156	157	⋯	151
158	155	156	157	160	157	157	162	⋯	154
161	157	158	157	159	156	156	157	⋯	153

158	159	163	157	158	155	163	159	⋯	156
154	154	156	156	159	155	156	159	⋯	148
152	155	155	156	158	155	157	160	⋯	147
154	155	156	155	156	154	160	154	⋯	153
155	158	160	155	159	160	156	158	⋯	153
159	158	157	157	160	160	158	157	⋯	153
155	158	156	158	159	157	156	158	⋯	151
156	159	157	157	154	154	159	160	⋯	152
152	156	150	153	155	159	153	158	⋯	151
155	156	153	153	158	156	159	160	⋯	147
160	158	157	156	158	157	156	158	⋯	149
157	156	160	160	158	159	161	164	⋯	152
157	157	157	161	160	161	160	161	⋯	151
155	156	157	158	157	156	158	158	⋯	152
160	156	158	163	157	157	156	160	⋯	150
153	159	156	158	160	157	162	159	⋯	152
158	160	160	163	159	159	160	160	⋯	149
159	163	165	160	160	162	166	164	⋯	152
162	160	160	163	160	163	165	166	⋯	148
157	160	156	162	160	160	164	162	⋯	149
160	160	160	161	160	160	163	160	⋯	147
162	160	161	161	160	159	160	160	⋯	149
159	161	156	163	161	161	161	163	⋯	143
160	161	164	165	162	161	162	163	⋯	151
165	161	164	162	162	164	162	165	⋯	148
161	161	165	160	160	163	164	161	⋯	149
165	162	160	160	160	162	162	165	⋯	145
161	161	163	162	162	163	166	161	⋯	145
160	161	157	159	161	160	161	160	⋯	146
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋱	⋮
121	126	130	162	102	160	172	133	⋯	113

LAMPIRAN C. Nilai *Pixel* dari Citra Rice pada Gambar 2.3

122	92	95	99	102	107	89	90	···	92
99	99	102	82	100	89	91	87	···	93
97	107	103	86	98	92	93	96	···	103
102	100	99	87	97	89	110	95	···	101
84	107	98	99	92	94	104	91	···	97
86	107	93	107	91	109	92	105	···	95
97	104	90	93	93	96	89	121	···	92
105	102	110	97	100	93	89	106	···	107
111	97	100	95	110	98	103	105	···	106
97	88	114	93	96	87	101	94	···	97
100	95	100	101	97	95	101	95	···	96
91	105	106	101	97	97	101	96	···	93
87	110	105	105	89	100	89	97	···	105
91	98	92	108	93	98	98	95	···	102
105	90	94	99	105	98	90	99	···	108
111	95	103	95	94	91	93	98	···	90
99	110	91	89	95	87	95	99	···	107
103	110	89	88	95	94	91	103	···	112
97	116	88	99	95	94	102	99	···	116
94	104	89	98	105	101	103	106	···	110
107	100	94	95	101	95	97	98	···	97
133	96	99	99	94	100	105	92	···	103
178	124	104	89	97	91	122	93	···	95
171	171	105	94	100	99	103	95	···	100
168	171	157	106	91	108	103	93	···	100
172	171	177	131	88	104	101	93	···	109
175	172	177	174	108	102	97	100	···	99
176	173	176	175	155	96	104	106	···	108
184	179	178	174	177	135	107	92	···	100

187	182	178	177	177	175	113	93	⋯	107
190	187	180	178	176	179	160	99	⋯	105
194	190	186	182	183	183	182	138	⋯	99
190	193	189	188	182	182	183	177	⋯	97
189	192	190	194	185	183	181	182	⋯	103
188	189	189	195	192	186	189	179	⋯	102
139	190	189	194	196	188	184	180	⋯	110
100	155	189	192	195	192	181	179	⋯	97
100	115	165	189	192	194	189	177	⋯	94
100	112	113	178	188	195	194	184	⋯	100
99	102	112	126	183	194	192	186	⋯	107
120	104	117	104	137	180	189	187	⋯	105
124	98	135	115	133	116	171	194	⋯	108
118	111	112	101	112	100	116	147	⋯	97
98	110	93	108	112	98	102	103	⋯	101
98	113	95	121	100	109	99	101	⋯	101
103	104	95	120	95	112	93	109	⋯	118
111	99	103	106	106	116	103	123	⋯	107
109	100	105	101	109	106	104	96	⋯	94
107	101	121	96	101	110	116	101	⋯	100
111	112	105	103	103	99	121	108	⋯	110
108	105	111	116	101	103	108	102	⋯	108
95	106	104	120	97	105	101	99	⋯	103
135	108	102	110	98	114	102	107	⋯	94
184	159	112	102	101	100	101	104	⋯	96
184	182	171	116	100	96	101	98	⋯	104
195	181	183	182	138	103	113	97	⋯	120
196	186	179	181	185	163	118	101	⋯	101
189	188	184	179	182	186	180	128	⋯	94
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋱	⋮
59	51	52	47	52	46	54	55	⋯	42

LAMPIRAN D. Skrip Program Enkripsi DNA-*Vigenere Cipher*

enkripdna_vigenere.m

```
function ciper=enkripdna_vigenere(plaintek,knc)
m=length(knc);
plain=double(plaintek);
for i=1:length(plain)
    binplain(i,:)=digit2biner(plain(i));
end
for i=1:m
    binknc(i,:)=digit2biner(knc(i));
end
kolomm=randi(8);
pla=konvtab(plain,binplain,kolomm);
kun=konvtab(knc,binknc,kolomm);
tbl=['ATCG';
'TCGA';
'CGAT';
'GATC'];
k='ATCG';
for i=1:length(pla)
for j=1:4
if pla(i)==k(j)
            brs=j;
break
end
end
for z=1:4
if kun(i)==k(z)
            kol=z;
break
```

```
end
end
    cip(i)=tbl(brs,kol);
end
ciper=([konvbin(cip,kolomm) kolomm]);
```

LAMPIRAN E. Skrip Program Dekripsi DNA-*Vigenere Cipher*

dekripdna_vigenere.m

```
function ciper=dekripdna_vigenere(plai,knc)
g=1;
for i=1:length(plai)/2
    heks=plai(g:g+1);
    plain(i)=hex2dec(heks);
    g=g+2;
end
for i=1:length(plain)
    binplain(i,:)=digit2biner(plain(i));
end
for i=1:length(knc)
    binknc(i,:)=digit2biner(knc(i));
end
kolomm=plain(length(plain));
pla=konvtab(plain,binplain,kolomm);
kun=konvtab(knc,binknc,kolomm);
k='ATCG';
tbl=['ATCG';
'TCGA';
'CGAT';
'GATC'];
for i=1:length(pla)
for z=1:4
if kun(i)==k(z)
            kol=z;
break
end
end
```

```
for j=1:4
if tbl(kol,j)==pla(i)
            brs=j;
break
end
end
    cip(i)=tbl(brs);
end
ciper=konvbin(cip,kolomm);
```