

Penerapan Algoritma Pencocokan *String* dalam Pendeteksian Mutasi DNA

Fathan Adi Pranaya 13511027
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
fathanpranaya@students.itb.ac.id

Abstrak—Mutasi adalah peristiwa perubahan struktur materi genetik pada suatu organisme atau makhluk hidup yang mengakibatkan terjadinya perubahan sifat atau karakter. Mutasi dapat berdampak positif maupun negatif. Makalah ini berisi tentang bagaimana mendeteksi sebuah mutasi yang terjadi pada suatu organisme dengan pendekatan ilmu komputasi. Metodenya adalah dengan memodelkan suatu DNA atau kromosom dari makhluk hidup menjadi sebuah teks atau *String* lalu mencocokkannya dengan pola-pola mutasi menggunakan algoritma *pattern matching* atau pencocokan *string*.

Kata kunci—algoritma, DNA, mutasi, *pattern matching*.

I. PENDAHULUAN

Saat ini teknologi semakin berkembang. Hampir semua persoalan yang ada di dunia ini mampu diselesaikan dengan bantuan teknologi. Manusia dapat memanfaatkan segala bentuk teknologi untuk mempermudah pekerjaannya, salah satunya adalah penggunaan teknologi radio aktif atau nuklir.

Teknologi radio aktif atau nuklir dapat dimanfaatkan di hampir semua bidang salah satunya adalah untuk pembangkit listrik atau sumber energi. Namun, dibalik semua manfaatnya itu radio aktif atau nuklir juga membawa dampak buruk bagi umat manusia salah satunya adalah digunakan untuk bom atom yang tidak hanya menyebabkan kerusakan tetapi juga membawa dampak mutasi pada organisme yang terkena radiasinya.

Sampai saat ini belum ada metode yang efektif untuk mendeteksi apakah seseorang terkena mutasi. Cara yang digunakan masih konvensional sehingga belum mampu menangani untuk skala yang lebih besar.

Dengan bantuan teknologi komputer saat ini semakin berkembang, pekerjaan manusia yang sifatnya berulang-ulang dengan jumlah yang sangat besar dapat direduksi. Sehingga pendeteksian mutasi dapat dilakukan oleh komputer.

Salah satu pendekatan yang dilakukan adalah dengan memodelkan DNA makhluk hidup menjadi sebuah teks atau *string* lalu mencocokkannya dengan pola-pola mutasi yang ada menggunakan algoritma pencocokan *string*.

II. MUTASI

Mutasi merupakan peristiwa perubahan struktur materi genetik pada suatu organisme atau makhluk hidup yang mengakibatkan terjadinya perubahan sifat atau karakter. Perubahan sifat mutasi tersebut dapat terjadi bila sel-sel yang terkena mutasi merupakan sel-sel gamet (mutasi germinal). Apabila peristiwa tersebut terjadi pada sel-sel tubuh (mutasi somatis), sifat baru yang didapat tidak akan diturunkan ke generasi berikutnya.

Istilah mutasi pertama kali digunakan oleh Hugo de Vries, seorang ahli botani Belanda ketika menjelaskan penyimpangan gen pada bunga *Oenothera lamarckiana*. Istilah lainnya yang berkaitan dengan mutasi, yakni faktor penyebab mutasi disebut mutagen, individu yang mengalami mutasi disebut mutan dan proses mutasi itu sendiri dinamakan mutagenesis.

Berdasarkan proses terjadinya, mutasi dapat terjadi secara spontan atau alami maupun buatan, yakni dengan campur tangan manusia. Sedangkan untuk ruang lingkungannya, mutasi dibedakan menjadi dua kategori, yaitu mutasi gen dan mutasi kromosom. Untuk lebih jelasnya, akan dibahas lebih rinci pada subbab berikutnya.

A. Mutasi Gen

Mutasi gen dapat disebut juga mutasi titik (point mutation) terjadi karena adanya perubahan struktur gen (DNA), yakni perubahan pada urutan nukleotida DNA yang meliputi perubahan jenis basa nitrogen dan perubahan jumlah basa nitrogen. Akibatnya, asam amino yang dikodekan berubah sehingga membentuk suatu protein yang salah.

Ada beberapa macam jenis mutasi gen, antara lain sebagai berikut.

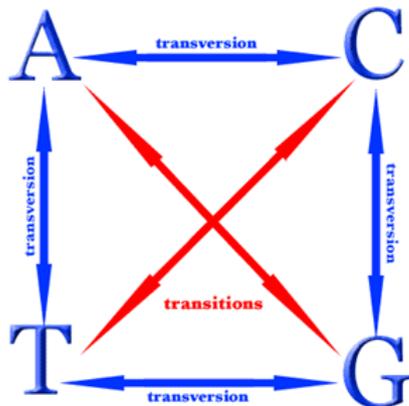
1. Delesi, merupakan jenis mutasi gen yang terjadi karena hilangnya satu atau beberapa basa nitrogen.
2. Addisi kebalikan dari delesi, yakni mutasi yang terjadi karena penambahan satu atau beberapa basa nitrogen sering disebut juga insersi.



Gambar 1. Addisi (kiri) dan Delesi (kanan) (images.google.com, 2013)

- Substitusi, merupakan pertukaran pasangan basa nitrogen pada DNA (A, G, C, T). Terdapat dua jenis basa nitrogen, yaitu basa nitrogen purin yang terdiri dari *adenine* (A) dan *guanine* (G), dan basa nitrogen pirimidin yang terdiri dari *cytosine* (C) dan *thymine* (T).

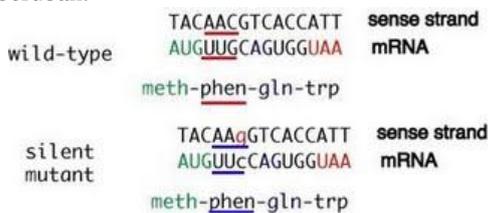
Berdasarkan perubahan jenis basanya substitusi dibedakan menjadi dua kategori, yaitu transisi dan transversi. Bila pertukaran terjadi antar pasangan basa nitrogen sesama purin (A-G) atau sesama pirimidin (C-T) disebut transisi. Misalnya pasangan AT digantikan dengan pasangan GC. Namun, bila pertukaran terjadi antara pasangan basa nitrogen purin dengan pirimidin dan sebaliknya (A/G-C/T), disebut transversi. Misalnya AT digantikan dengan pasangan TA.



Gambar 2. jenis substitusi pasangan basa nitrogen, transisi dan transversi (images.google.com, 2013)

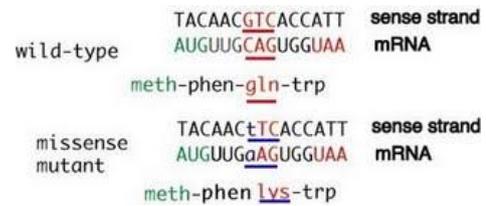
Adapun dampak dari mutasi gen ini dibedakan menjadi tiga kelompok, yaitu

- Mutasi diam, yakni jika terjadi perubahan kode genetik tetapi tidak sampai mengakibatkan perubahan pada asam amino yang disintesis. Sehingga fenotip dari organisme tersebut juga tidak berubah.



Gambar 3. ilustrasi mutasi diam (images.google.com, 2013)

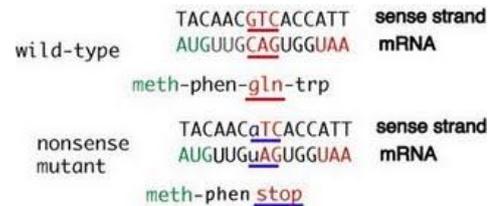
- Mutasi salah arti, yaitu jika terjadi perubahan kode genetik dan mengakibatkan perubahan asam amino yang disintesis sehingga fenotip berubah.



Gambar 3. Mutasi salah arti (images.google.com, 2013)

Pada gambar di atas, terjadi perubahan kode genetik (mRNA) CAG menjadi aAG (transversi C-A) sehingga terjadi perubahan pada asam amino yang dihasilkan, yakni dari glutamin (gln) menjadi lisin (lys).

- Mutasi tanpa arti, jika terjadi perubahan kode genetik yang mengkodekan kodon stop (akhir dari rantai asam amino), sehingga protein yang dihasilkan biasanya non-fungsional dan tidak berpengaruh terhadap fenotip dari organisme tersebut. Contohnya adalah seperti pada gambar di bawah ini. Protein yang seharusnya terbentuk pada rantai asam amino ini adalah protein yang mengandung meth-phen-gln-trp, tetapi terjadi mutasi dan mengakibatkan protein yang terbentuk hanya mengandung asam amino meth dan phen.



Gambar 4. Mutasi tanpa arti (images.google.com, 2013)

B. Mutasi Kromosom

Mutasi kromosom (aberasi/*gross mutation*) dapat disebabkan karena perubahan struktur kromosom maupun perubahan jumlah kromosom.

- Perubahan struktur kromosom

Mutasi karena perubahan struktur kromosom berlangsung secara spontan dan biasanya disebabkan karena induksi bahan kimia atau radiasi radio aktif. Perubahan ini dapat dilihat ketika proses pembelahan sel baik itu mitosis (sel tubuh) maupun meiosis (sel kelamin).

Mutasi kromosom yang terjadi karena perubahan struktur kromosom meliputi, delesi (pengurangan), inversi (pembalikan), duplikasi (penggandaan), translokasi (pemindahan) dan katenasi.

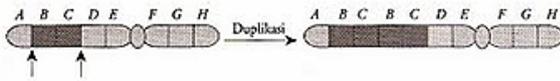
Delesi merupakan perubahan struktur kromosom yang terjadi karena hilangnya sebagian segmen kromosom yang

mengandung kode genetik karena patah atau rusak selama pembelahan sel.



Gambar 5. Delesi Kromosom
(images.google.com, 2013)

Duplikasi merupakan perubahan struktur kromosom karena patahnya sebagian segmen kromosom, lalu patahan tersebut tersambung pada kromosom homolog. Sehingga mengakibatkan munculnya duplikasi segmen dari kromosom yang patah tersebut.



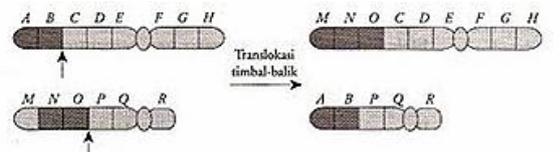
Gambar 6. Duplikasi Kromosom
(images.google.com, 2013)

Inversi merupakan patahnya sebagian segmen kromosom, lalu patahannya tersambung kembali dengan kromosom tersebut, namun posisinya terbalik. Ada dua macam inversi, yaitu inversi perisentrik bila melibatkan perubahan posisi sentromer (titik tengah pada kromosom) dan inversi parasentrik bila tidak melibatkan posisi setromer.



Gambar 7. Inversi Kromosom
(images.google.com, 2013)

Translokasi adalah patahnya sebagian segmen kromosom lalu patahan tersebut tersambung pada kromosom lain yang tidak homolog. Ada dua jenis translokasi, yaitu translokasi resiprok (timbang balik) dan translokasi nonrespirok.



Gambar 8. Translokasi Kromosom
(images.google.com, 2013)

Yang terakhir, katenasi merupakan translokasi dari dua buah kromosom non-homolog yang menyebabkan dua pasang kromosom tersebut membentuk struktur menyerupai lingkaran.

2. Perubahan jumlah kromosom

Mutasi karena perubahan jumlah kromosom digolongkan menjadi euploidi (secara keseluruhan) dan aneuploidi (hanya sebagian).

Euploidi merupakan perubahan set kromosom secara keseluruhan. Bila umumnya set kromosom manusia normal adalah $2n$ (diploid) berubah menjadi n (monoploid), $3n$ (triploid) dan $4n$ (tetraploid). Contoh mutasi ini adalah pada buah semangka tanpa biji.

Aneuploidi merupakan perubahan sebagian pasangan kromosom baik hilang maupun berlebih. Beberapa jenis aneuploidi antara lain,

- Nulisomi, kehilangan sepangan kromosom
- Monosomi, kehilangan satu buah kromosom
- Trisomi, kelebihan satu buah kromosom
- Tetrasomi, kelebihan sepasang kromosom

III. ALGORITMA *PATTERN MATCHING*

Pattern matching atau biasa dikenal dengan *String Matching*/Pencocokan *string* merupakan sebuah algoritma mencocokkan dua buah *string*, yaitu *string* teks (T) dan *string pattern*/pola (P). Misalnya diberikan sebuah teks “nobody but you” dan diberikan *pattern* “but”. Artinya algoritma pencocokan *string* akan mencari apakah pola tersebut terdapat pada teks yang diberikan.

Hingga saat ini sudah banyak algoritma pencocokan *string* yang dikembangkan. Tetapi dalam makalah ini, penulis hanya akan membahas mengenai algoritma yang diajarkan pada perkuliahan saja, yakni *brute force*, Knuth-Morris-Pratt (KMP) dan Boyer-Moore (BM). Untuk lebih jelasnya, akan dijelaskan pada subbab berikut ini.

A. Algoritma *Brute force*

Algoritma *brute force*, sesuai dengan namanya algoritma ini menggunakan pendekatan yang paling sederhana (*straightforward*), langsung dan dengan cara yang jelas (*obvious way*) dalam memecahkan masalah. Karena kesederhanaan dan cara yang jelas itu lah sering kali algoritma ini membutuhkan *resource* yang lebih daripada algoritma lain dalam pemecahan masalahnya. Sehingga algoritma ini sering dianggap tidak mangkus dan kurang tepat untuk digunakan pemecahan masalah dengan skala yang besar.

Seperti yang diperlihatkan sebelumnya, *brute force* pada pencocokan *string* mempunyai algoritma sebagai berikut.

1. Mula-mula pola P dicocokkan dengan awal teks T
2. Kemudian perbandingan karakter bergerak dari kiri ke kanan setiap karakter yang bersesuaian sampai
 - a. semua karakter yang dibandingkan pada pola P cocok (pencarian berhasil)
 - b. terdapat karakter yang tidak sama
3. Bila b terjadi maka geser pola P satu karakter ke kanan dan ulangi langkah 2.

Teks :	nobody but you
Pola :	but
	nobody but you
s=0 :	but
s=1 :	but
s=2 :	but
s=3 :	but
s=4 :	but
s=5 :	but
s=6 :	but
s=7 :	but

Gambar 9. Pencocokan *string* dengan *brute force*

Ilustrasi di atas merupakan salah satu contoh sederhana proses pencocokan *string* dengan menggunakan algoritma *brute force*.

Untuk kompleksitas algoritmanya dihitung berdasarkan jumlah pencocokan karakter yang terjadi. Pada kasus terbaiknya, algoritma *brute force* memiliki kompleksitas $O(n)$ yaitu kasus di mana karakter pertama dari pola P tidak pernah sama dengan karakter pada teks T kecuali pada saat pencocokan terakhir. Sedangkan untuk kasus terburuknya, adalah ketika semua karakter kecuali karakter terakhir pada pola P sama dengan karakter pada teks T. Sehingga kompleksitas untuk kasus ini adalah $O(mn)$.

B. Algoritma Knuth-Morris-Pratt

Algoritma Knuth-Morris-Pratt atau biasa disebut KMP merupakan varian dari algoritma pencocokan *string*. Sama dengan *brute force*, algoritma ini juga menggunakan mekanisme perbandingan karakter pada pola dengan karakter pada teks. Perbedaannya dibandingkan dengan *brute force* adalah dari segi jumlah pergeseran yang dilakukan ketika terdapat karakter pada pola yang tidak sama dengan karakter pada teks.

Algoritma KMP memelihara informasi yang digunakan untuk menghitung jumlah pergeseran yang seharusnya dilakukan. Informasi pergeseran tersebut didapatkan dari fungsi yang dinamakan *Border Function* atau fungsi pinggiran.

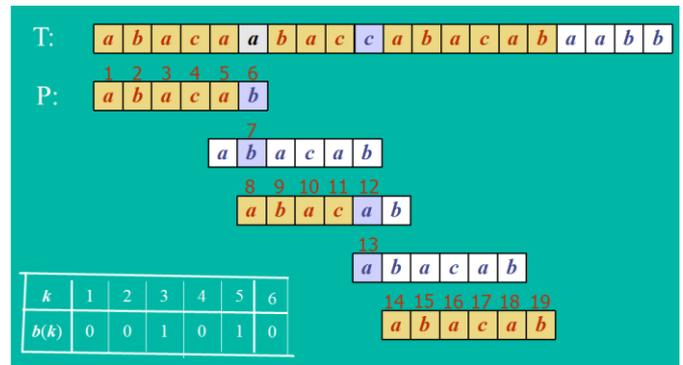
Dengan adanya fungsi pinggiran ini, pergeseran-pergeseran yang tidak berguna dapat dikurangi sehingga membuat algoritma ini lebih cepat dan lebih mangkus dibandingkan dengan *brute force*. Untuk lebih jelasnya, ilustrasi cara kerja algoritma KMP adalah sebagai berikut.

Diberikan teks = abacaabaccabacabaabb dan sebuah pola = abacab. Sebelum memulai proses pencocokan *string*, akan dilakukan perhitungan jumlah pergeseran dengan menggunakan border function. Dari pola yang diberikan, didapat hasil border function sebagai berikut.

k	1	2	3	4	5	6
b(k)	0	0	1	0	1	0

Tabel 1. Border Function algoritma KMP (Munir, 2013)

Setelah perhitungan border function, barulah dimulai proses pencocokan *string*. Pada pencocokan pertama kali terlihat karakter pada pola *match* dengan karakter pada teks dari awal sampai dengan karakter dengan indeks k=5. Sehingga berdasarkan border function, pergeseran yang harus dilakukan adalah sebanyak k-b(k), yaitu 4 karakter ke kanan dan pencocokan karakter mulai dari karakter k=2 yaitu karakter b dan seterusnya.



Gambar 10. Pencocokan string dengan algoritma KMP (Munir, 2013)

C. Algoritma Boyer-Moore

Sama halnya dengan KMP, algoritma Boyer-Moore atau biasa disebut BM ini merupakan salah satu varian dari algoritma pencocokan *string*. Perbedaan yang mencolok bila dibandingkan dengan dua algoritma sebelumnya adalah urutan perbandingan karakternya.

Pada algoritma *brute force* dan KMP, perbandingan karakter pola dengan teks dilakukan dari karakter paling kiri ke kanan. Tetapi pada algoritma BM, perbandingan karakter pola dengan teks dimulai dari karakter paling kanan.

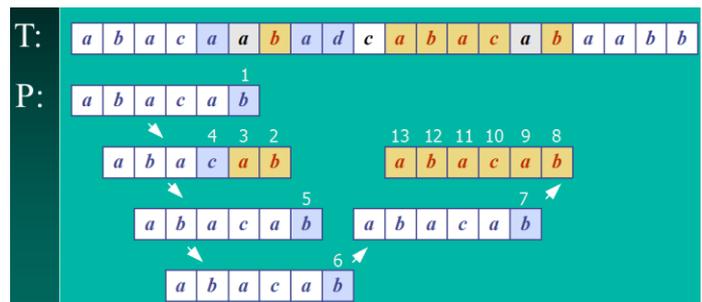
Sama halnya dengan KMP, algoritma BM menyimpan informasi berupa tabel indeks kemunculan terakhir (*last occurrence*) karakter pada pola yang digunakan untuk mekanisme pergeserannya. Untuk lebih jelasnya, ilustrasi algoritma BM adalah sebagai berikut.

Diberikan teks=abacaabaccabacabaabb dan pola=abacab. Diperoleh informasi *last occurrence* dari pola adalah sebagai berikut.

x	a	b	c	d
L(x)	5	6	4	-1

Tabel 2. Informasi Last Occurrence algoritma BM (Munir, 2013)

Nilai *last occurrence* karakter d diberi -1 karena karakter d tidak terdapat pada pola. Sehingga proses pencocokan *string* dengan algoritma BM adalah sebagai berikut.



Gambar 11. Pencocokan string dengan algoritma BM (Munir, 2013)

IV. PENDETEKSIAN MUTASI DENGAN ALGORITMA PATTERN MATCHING

Seperti yang telah dijelaskan sebelumnya, struktur pembentuk DNA atau kromosom merupakan kumpulan dari basa nitrogen baik itu purin maupun pirimidin yang membentuk suatu rantai yang berurutan. Sehingga DNA atau kromosom dari suatu organisme dapat dimodelkan menjadi sebuah teks atau *string*.

Dengan pendekatan ini, memungkinkan kita untuk mendeteksi suatu mutasi baik itu pada level DNA maupun kromosom dengan pendekatan *pattern matching* atau pencocokan *string*. Pada pendeteksian ini, DNA atau kromosom dianggap sebagai sebuah teks sedangkan jenis-jenis kelainan yang disebabkan oleh mutasi dapat dimodelkan menjadi sebuah pola atau *pattern*.

Jika suatu pola mutasi tersebut ditemukan pada teks yang merupakan representasi dari DNA atau kromosom, maka dapat dikatakan bahwa organisme tersebut telah mengalami mutagenesis atau bisa disebut sebagai mutan.

Karena belum ada standar yang baku dalam memodelkan suatu struktur DNA atau kromosom organisme menjadi sebuah teks/*string* dan belum ada pola-pola yang umum dari mutasi-mutasi yang dijelaskan sebelumnya maka penulis mencoba memodelkan DNA atau kromosom menjadi sebuah teks dengan pendekatan penulis sendiri. Sementara untuk pola-pola mutasi, penulis mengambil contoh mutasi yang spesifik yang terdapat pada literature lain.

A. Pemodelan DNA dan Pola Mutasi

Untuk proses mendeteksi mutasi pada gen, penulis menggunakan sampel DNA yang menyebabkan β -thalassaemia.

Teks DNA :

```
TCCTAAGCCA GTGCCAGAAG AGCCAAGGAC
AGGTACGGCT GTCATCACTT AGACCTCACC
CTGTGGAGCC ACACCCTAGG GTTGGCCAAT
CTACTCCCAG GAGCAGGGAG GGCAGGAGCC
AGGGCTGGGC ATAAAAGTCA GGGCAGAGCC
ATCTATTGCT TACATTTGCT TCTGACACAA
CTGTGTTTAC TAGCAACCTC AAACAGACAC
CATGGTGCAC CTGACTCCTG AGGAGAAGTC
TGCCGTTACT GCCCTGTGGG GCAAGGTGAA
CGTGGATGAA GTTGGTGGTG AGGCCCTGGG
CAGGTTGGTA TCAAGGTTAC AAGACAGGTT
TAAGGAGACC AATAGAAACT GGGCATGTGG
AGACAGAGAA GACTCTTGGG TTTCTGATAG
GCACTGACTC TCTCTGCCTA TTGGTCTATT
TTCCACCCCT TAGGCTGCTG GTGGTCTACC
CTTGGACCCA GAGGTTCTTT GAGTCCTTTG
GGGATCTGTC CACTCCTGAT GCTGTTATGG
GCAACCCTAA GGTGAAGGCT CATGGCAAGA
AAGTGCTCGG TGCCTTTAGT GATGGCCTGG
CTCACCTGGA CAACCTCAAG GGCACCTTTG
CCACTGAG TGAGCTGCAC TGTGACAAGC
TGCACGTGGA TCCTGAGAAC TTCAGGGTGA
```

```
GTCTATGGGA CCCTTGATGT TTTCTTTCCC
CTTCTTTTCT ATGGTAAAGT TCATGTCATA
GGAAGGGGAG AAGTAACAGG GTACAGTTTA
GAATGGGAAA CAGACGAATG ATTGCATCAG
```

Gambar 12. Hasil pemodelan DNA menjadi teks/*string*
(ScienceAsia.org, 2011)

Berdasarkan penjelasan sebelumnya, mutasi gen dapat terjadi karena addisi, delesi, substitusi sebuah basa nitrogen. Jika kita mengambil pola dengan hanya satu karakter, maka kita tidak dapat membedakan mana basa purin atau pirimidin yang normal dan yang abnormal. Sehingga penulis mengambil pola dengan panjang 6 karakter walaupun peristiwa mutasi (transversi) terjadi hanya pada salah satu karakter tersebut. Dengan demikian pola untuk pencocokan *string* adalah sebagai berikut.

Jenis mutasi (gen) : Transversi, GTTGCT -> TTTGCT (Perubahan basa nitrogen guanine menjadi timin).

Pola pencocokan *string* : TTTGCT. Kemudian akan dicari pola TTTGCT dalam teks DNA di atas.

B. Hasil

```
C:\Windows\system32\cmd.exe
C:\Users\Fathan\workspace\Mutasi\src>java Pa
DNA :
TCCTAAGCCA GTGCCAGAAG AGCCAAGGAC
AGGTACGGCT GTCATCACTT AGACCTCACC
CTGTGGAGCC ACACCCTAGG GTTGGCCAAT
CTACTCCCAG GAGCAGGGAG GGCAGGAGCC
AGGGCTGGGC ATAAAAGTCA GGGCAGAGCC
ATCTATTGCT TACATTTGCT TCTGACACAA
CTGTGTTTAC TAGCAACCTC AAACAGACAC
CATGGTGCAC CTGACTCCTG AGGAGAAGTC
TGCCGTTACT GCCCTGTGGG GCAAGGTGAA
CGTGGATGAA GTTGGTGGTG AGGCCCTGGG
CAGGTTGGTA TCAAGGTTAC AAGACAGGTT
TAAGGAGACC AATAGAAACT GGGCATGTGG
AGACAGAGAA GACTCTTGGG TTTCTGATAG
GCACTGACTC TCTCTGCCTA TTGGTCTATT
TTCCACCCCT TAGGCTGCTG GTGGTCTACC
CTTGGACCCA GAGGTTCTTT GAGTCCTTTG
GGGATCTGTC CACTCCTGAT GCTGTTATGG
GCAACCCTAA GGTGAAGGCT CATGGCAAGA
AAGTGCTCGG TGCCTTTAGT GATGGCCTGG
CTCACCTGGA CAACCTCAAG GGCACCTTTG
CCACTGAG TGAGCTGCAC TGTGACAAGC
TGCACGTGGA TCCTGAGAAC TTCAGGGTGA

Pola mutasi : TTTGCT

-----
Pencocokkan String dengan algoritma KMP :
Jenis mutasi : Transversi Basa Nitrogen
Jumlah iterasi : 378

-----
Pencocokkan String dengan algoritma BM :
Jenis mutasi : Transversi Basa Nitrogen
Jumlah iterasi : 76

C:\Users\Fathan\workspace\Mutasi\src>
```

Gambar 13. Hasil pencocokan *string* DNA dengan pola mutasi

Berdasarkan keluaran program di atas, didapatkan informasi bahwa pola mutasi (transversi basa nitrogen) terdeteksi pada DNA yang telah dimodelkan menjadi *string* (lihat gambar 12).

Dengan menggunakan algoritma KMP terhitung jumlah iterasi adalah sebanyak 378 kali. Sedangkan dengan menggunakan algoritma BM jumlah iterasi yang dilakukan adalah 76 kali.

V. KESIMPULAN

Dengan demikian terbukti bahwa pendeteksian mutasi DNA dapat dilakukan dengan pendekatan ilmu komputasi, yaitu dengan algoritma *pattern matching* atau pencocokan *string*.

Dari hasil keluaran program, terbukti bahwa algoritma BM menghasilkan jumlah iterasi yang lebih sedikit dibandingkan dengan jumlah iterasi pada algoritma KMP. Sehingga algoritma BM merupakan algoritma yang lebih cocok digunakan untuk mendeteksi pola mutasi pada DNA daripada algoritma KMP maupun *brute force*.

REFERENSI

- [1] M. Rinaldi, Diktat Kuliah IF2211 Strategi Algoritma ,” Teknik Informatika Institut Teknologi Bandung”, Bandung, 2009, pp. 186–193.
- [2] Nopparatana C, Fukumaki Y (1995) The spectrum of b-thalassemia mutations in southern Thailand. Southeast Asian J Trop Med Publ Health 26, [suppl 1], 229–34.
- [3] http://www.bioinformatics.org/sms2/mutate_dna.html 8:30AM 17/12/2013
- [4] <http://biologimediacentre.com/mutasi/> 9:30AM 17/12/2013
- [5] http://evolution.berkeley.edu/evolibrary/article/mutations_01 11:30AM 17/12/2013
- [6] <http://images.google.com/> 12:00 AM 17/12/2013

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 18 Desember 2013



Fathan Adi Pranaya - 13511027