

KONSEP ALGORITMA GENETIKA SEBAGAI DASAR PEMBUATAN TEKA TEKI SILANG BERBASIS KOMPUTER

Dini M.Hutagalung

mhdini@gmail.com
Universitas Sari Mutiara Indonesia

ABSTRAK

Pada era millenium ini, hampir semua kegiatan kehidupan manusia dilakukan dengan menggunakan teknologi. Termasuk dalam bidang pendidikan. Salah satu cara yang cukup efektif adalah dengan menggunakan Teka-Teki Silang (TTS). Di masa millenium ini, adalah wajar menggunakan teknologi informasi untuk melaksanakan suatu ujian, yang terkenal dengan nama CBT (Computer Test based) yang mana lebih menarik jika dibandingkan dengan menggunakan cara tradisional yaitu PTB (Paper Test Based). Dengan mengeksplor konsep daripada algoritma dan menuangkannya ke dalam aplikasi Visual Basic, penulis mencoba membuat Teka Teki Silang dengan menggunakan komputer. Penulis menggunakan algoritma Genetika sebagai konsep pengkodean untuk mengkonversikan kombinasi pertanyaan secara mendatar dan menurun. Semakin banyak jumlah pertanyaan, maka semakin banyak kombinasi yang harus dicoba sehingga membutuhkan waktu yang lama. Algoritma genetika adalah algoritma pencarian berdasarkan mekanisme seleksi alam dan genetik. Algoritma ini dapat digunakan untuk menyusun kombinasi dari sejumlah jawaban pertanyaan dalam pembentukan TTS. Waktu yang diperlukan untuk menyusun TTS dengan menggunakan algoritma genetika cukup singkat (beberapa detik). Dengan menggunakan bahasa pemrograman Visual BASIC 6.0 dapat dibuat program aplikasi penyusunan TTS dengan algoritma genetika.

Kata Kunci: Algoritma, Visual Basic, TTS, Genetika, Visual Basic

I. PENDAHULUAN

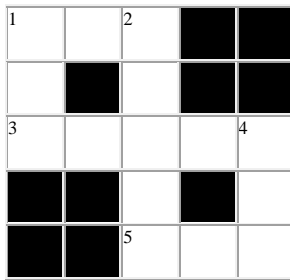
Teka Teki Silang (TTS) merupakan sebuah permainan untuk mengasah otak. TTS dapat dimanfaatkan dalam bidang pendidikan untuk menguji kemampuan anak didik, sehingga pertanyaan-pertanyaan yang digunakan disesuaikan dengan kebutuhan saja. Untuk membuat sebuah teka-teki silang dapat dikatakan mudah, mudah karena hanya mengkombinasikan kata yang satu dengan yang lain. Tetapi akan sulit apabila jumlah pertanyaan yang digunakan banyak, karena untuk membuat sebuah teka-teki silang dari sekumpulan pertanyaan, maka harus

dicoba kombinasi yang mungkin dari kata yang ada sampai terbentuk sebuah TTS. Semakin banyak jumlah pertanyaan, maka semakin banyak kombinasi yang harus dicoba. Dalam optimasi, penerapan algoritma genetika digunakan untuk mendapatkan suatu pola penyusunan kata dalam ruang-ruang kosong yang paling optimal. Untuk mengetahui optimal atau tidaknya pengisian ruang kosong dilakukan dengan melihat banyaknya kombinasi kata yang dapat dibuat, atau dapat juga dilihat dari banyaknya sisa ruang kosong teka-teki silang. Jadi, semakin banyak kombinasi kata yang dibuat untuk mengisi ruang-ruang

kosong, maka semakin optimal pengisian ruang kosong TTS. Solusi ini dapat diatasi apabila membuat teka-teki silang dilakukan secara otomatis dengan komputer.

II. LANDASAN TEORI

Teka-teki silang atau disingkat TTS adalah suatu permainan yang mengharuskan penggunanya untuk mengisi ruang-ruang kosong dengan huruf-huruf yang membentuk sebuah kata berdasarkan petunjuk yang diberikan (Wikipedia, 2010). Petunjuk biasa dibagi dalam kategori mendatar dan menurun tergantung posisi kata yang harus diisi.



PERTANYAAN	
Mendatar	
1. Kuat	
3. Giliran	
5. Tanpa	
Menurun	
1. Akademi Bahasa Asing	
2. Perhiasan	
4. Elektron	

Gambar II.1 Contoh sebuah TTS sederhana

A. Algoritma Genetika

Algoritma genetika adalah suatu algoritma pencarian (*searching*) berdasarkan cara kerja melalui mekanisme seleksi alam dan genetik. Tujuannya untuk menentukan struktur-struktur yang disebut individu berkualitas tinggi di dalam suatu domain yang disebut populasi untuk mendapatkan solusi persoalan (Pandjaitan, Lanny W., 2007 : 163).

Pada tahun 1975, John Holland memperkenalkan algoritma genetika untuk yang pertama kalinya. Algoritma genetika berbeda dengan algoritma konvensional karena dimulai dari suatu himpunan awal yang disebut populasi. Algoritma genetika menggunakan dua prinsip dasar dalam sistem biologis, yaitu seleksi terhadap spesies yang ada dan peningkatan keanekaragaman (gen dengan operasi genetik) (Desiani, Anita, 2006 : 187).

B. Pengkodean kromosom

Pengkodean merupakan bagian yang penting dalam menyelesaikan suatu permasalahan dengan algoritma genetika. Pengkodean adalah pengkonversian masalah dalam dunia nyata menjadi bentuk yang dapat diolah dengan menggunakan algoritma genetika. Pengkodean yang tepat sangat menentukan berhasil atau tidaknya proses algoritma genetika dalam menyelesaikan sebuah permasalahan. Pengkodean yang

tepat juga akan menentukan tingkat efisiensi komputasi yang digunakan (Kuswadi, Son, 2007 : 180).

Ada beberapa jenis pengkodean yang dapat digunakan dalam algoritma genetika, diantaranya pengkodean biner (*binary encoding*) dan pengkodean permutasi (*permutation encoding*).

C. Pengkodean biner (*binary encoding*)

Pengkodean biner merupakan pengkodean yang sering digunakan dan paling sederhana. Sesuai dengan namanya, pada pengkodean biner nilai gen pada sebuah kromosom hanya terdiri dari 0 dan 1 (Desiani, Anita, 2006 : 192). Contoh kromosom yang menggunakan pengkodean biner ditunjukkan pada gambar II.2.

Kromosom 1	1 0 1 0 1 0 1 0 1 1
Kromosom 2	1 1 0 0 0 0 1 0 1 0

Gambar II.2 Kromosom dengan pengkodean biner

(Sumber : Desiani, Anita, 2006 : 198)

D. Pengkodean permutasi

Pengkodean permutasi dapat digunakan untuk menyelesaikan *ordering problem*, seperti *traveling salesman problem* dan masalah penjadwalan. Dalam pengkodean permutasi urutan atau posisi gen pada kromosom merepresentasikan urutan dari suatu proses. Dalam pengkodean permutasi tidak ada nilai gen yang

sama dalam suatu kromosom. Contoh kasus yang dapat diselesaikan dengan pengkodean permutasi adalah TSP (*Traveling Salesman Problem*). *Traveling salesman problem* merupakan cara bagaimana menentukan rute minimal untuk mengunjungi beberapa tempat atau kota (Kuswadi, Son, 2007 : 177).

Jika ada 10 kota yang akan dikunjungi, maka satu kromosom akan mempunyai 10 gen, dimana tiap-tiap gen mewakili satu kota. Posisi atau urutan gen dalam kromosom menentukan urutan kota yang harus dilalui. Contoh kromosomnya dapat dilihat pada gambar II.3.

Kromosom 1	2 6 7 5 1 3 4 9 8 10
Kromosom 2	10 5 4 9 7 1 3 2 6 8

Gambar II.3 Kromosom dengan pengkodean permutasi

(Sumber : Desiani, Anita, 2006 : 198)

Pada kromosom 1 berarti kota yang pertama dikunjungi adalah kota 1, kemudian kota 2 demikian seterusnya sampai kota 10.

E. Parameter algoritma genetika

Dalam penerapan algoritma genetika, ada beberapa parameter yang dilibatkan, dimana parameter ini menentukan kesuksesan suatu proses optimasi. Jenis

parameter yang digunakan bergantung pada permasalahan yang diselesaikan (Kuswadi, 2007: 173).

F. Ukuran populasi

Ukuran populasi adalah jumlah kromosom yang ada pada populasi. Memilih ukuran populasi yang tepat akan meningkatkan kinerja algoritma genetika. Apabila ukuran populasi terlalu kecil, maka algoritma genetika hanya mempunyai alternatif solusi yang sedikit. Namun, jika ukuran populasi terlalu, besar proses algoritma genetika akan berjalan lambat.

a) Jumlah generasi

Jumlah generasi adalah jumlah *epoch* yang dijalani untuk satu rangkaian proses algoritma genetika. Satu proses rangkaian algoritma genetika dimulai dari proses seleksi, *crossover*, mutasi sampai *update* generasi. Proses algoritma genetika akan dihentikan apabila jumlah generasi sudah terpenuhi. Solusi yang diambil adalah kromosom dengan nilai *fitness* terbaik dari generasi terakhir.

b) Grafika Komputer

TTS ditampilkan di layar komputer dengan menggunakan teknik pengolahan grafik. Bidang ilmu yang khusus menangani pengolahan grafik adalah grafika komputer. Grafika komputer (*Computer Graphics*) adalah sebuah disiplin ilmu yang mempelajari proses untuk menciptakan suatu gambar

berdasarkan deskripsi objek maupun latar belakang yang terkandung pada gambar tersebut. Hal ini meliputi teknik-teknik untuk membuat gambar objek sesuai dengan keberadaan objek tersebut di alam nyata. Grafika komputer mencoba memvisualisasikan suatu informasi menjadi citra. Jadi, input dari grafika komputer adalah informasi atau data deskriptif tentang citra yang akan digambar, sedangkan outputnya adalah berupa citra (Sutoyo *et al*, 2009 : 5).

Grafika komputer bertujuan menghasilkan citra (lebih tepat disebut grafik atau *picture*) dengan primitif-primitif geometri seperti garis, lingkaran dan sebagainya. Primitif-primitif geometri tersebut memerlukan data deskriptif untuk melukis elemen-elemen gambar. Contoh data deskriptif adalah koordinat titik, panjang garis, jari-jari lingkaran, tebal garis, warna dan sebagainya. Grafika komputer memainkan peranan penting dalam visualisasi dan *virtual reality* (Munir, R, 2004 : 4).

Grafika komputer lebih menitikberatkan pada segi rancang bangun yang berorientasi geometri. Kenyataannya, gambar yang dihasilkan oleh grafika komputer memiliki tampilan yang tidak berbeda jauh dari objek sebenarnya. Blok diagram grafika komputer diperlihatkan pada Gambar II.15.

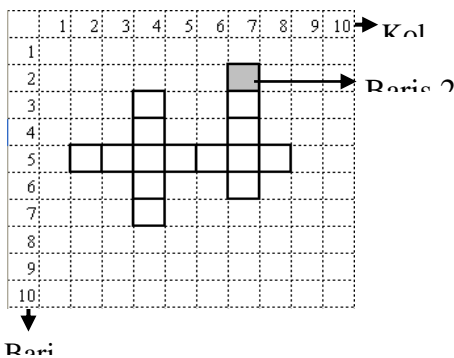


Gambar II.15 Bidang Studi Grafika Komputer

c) Menggambar Kotak

Secara umum, isian jawaban TTS ditampilkan dalam bentuk susunan kotak. Untuk memudahkan deskripsi lokasi penempatan kotak, seluruh area (ruang kerja) diberi nomor baris dan nomor kolom.

Dengan menggunakan prinsip kerja seperti disebutkan di atas, maka untuk menampilkan sebuah kotak pada lokasi tertentu dapat digunakan suatu deskripsi *DrawCell (row, col)*. Misalnya untuk menampilkan kotak pada baris 2 kolom 7, dapat digunakan deskripsi *DrawCell (2, 7)*. *DrawCell* adalah perintah yang dirancang sendiri oleh pembuat program yakni berupa subprogram yang digunakan untuk menggambar kotak, seperti terlihat pada Gambar II.17.



**Gambar II.17 Penomoran lokasi kotak TTS
Sekilas Tentang Visual Basic**

Visual Basic adalah bahasa pemrograman yang digunakan untuk membuat aplikasi berbasis grafis

(GUI *Graphical User Interface*). Visual Basic termasuk bahasa pemrograman yang berorientasi objek (*Object Oriented Programming*), artinya suatu program aplikasi yang disusun berdasarkan objek-objek. Objek berarti sesuatu yang dapat diamati. Dalam Visual Basic terdapat empat hal yang penting diperhatikan, yaitu *Object, Property, Event, dan Method*.

Multiple Document Interface (MDI) adalah fasilitas yang disediakan oleh Visual Basic, yang memungkinkan pemrogram untuk membuat sejumlah form di dalam satu form induk. Dalam hal ini *user* dapat membuka beberapa dokumen pada waktu yang bersamaan dalam satu form induk. Masing-masing dokumen yang dibuka ditampilkan form/window tersendiri dan disebut form anak (*childform*).

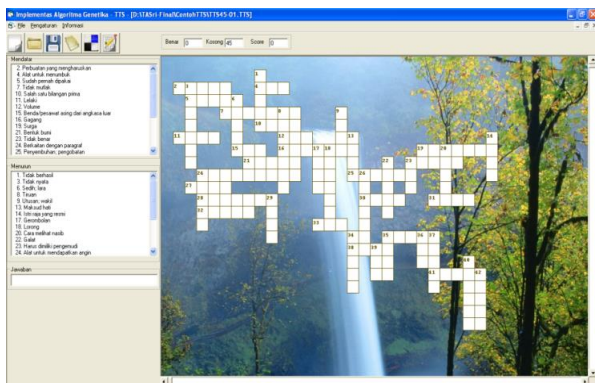
III. IMPLEMENTASI

Aplikasi Penyusunan TTS dengan Algoritma Genetika ini dibangun dengan menggunakan Visual BASIC 6.0, karena bahasa program ini dapat digunakan untuk membuat aplikasi secara cepat dan mudah. Visual BASIC 6.0 menyediakan berbagai fasilitas untuk penanganan grafis, fasilitas untuk membuat *user interface* yang menarik serta instruksi penanganan file.

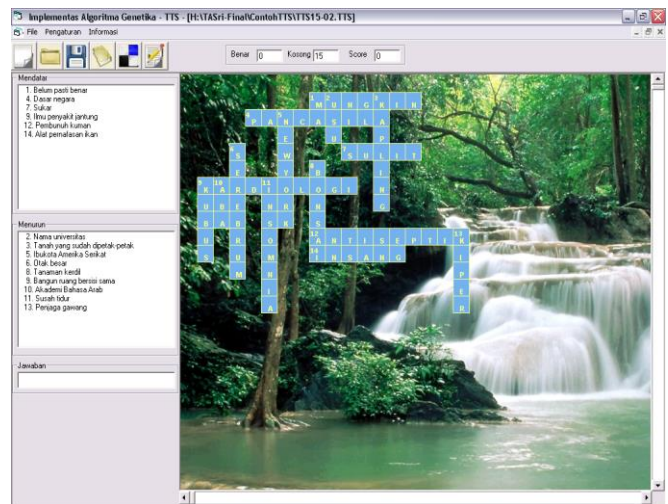
Penyusunan TTS dengan Algoritma Genetika ini dibangun dengan menggunakan Visual BASIC 6.0, karena bahasa program ini dapat digunakan untuk membuat aplikasi secara cepat dan mudah. Visual BASIC 6.0 menyediakan berbagai fasilitas untuk penanganan grafis, fasilitas untuk membuat *user interface* yang menarik serta instruksi penanganan file.

A. Jendela Teka Teki Silang

Jendela Teka Teki Silang merupakan implementasi dari rancangan *form* TTS. Di sisi kiri jendela ditampilkan soal-soal pertanyaan mendatar dan menurun, dan sisi kanan jendela ditampilkan kotak TTS yang siap untuk diisi oleh pengguna. Jendela Teka Teki Silang dengan jumlah soal 45 diperlihatkan pada Gambar IV. 5. Jendela Teka Teki Silang dengan jumlah soal 15 diperlihatkan pada Gambar IV. 6. Jendela Teka Teki Silang dengan jumlah soal 5 diperlihatkan pada Gambar IV. 7.



Gambar IV.5 Jendela Permainan TTS dengan jumlah soal 45



Gambar IV.6 Jendela Permainan TTS dengan jumlah soal 15

IV. PENUTUP

A. Kesimpulan

Dari hasil pengujian yang dilakukan dapat disimpulkan bahwa:

1. Algoritma genetika dapat diterapkan untuk menyusun Tekan Teki Silang. Penyusunan Tekan Teki Silang dapat dilakukan dalam tempo beberapa detik saja.
2. Terkadang aplikasi tidak berhasil membuat teka-teki silang sesuai dengan jumlah pertanyaan seperti yang diinputkan pengguna. Hal ini dikarenakan algoritma genetika tidak berhasil menemukan solusi sesuai dengan soal yang terpilih secara acak. Kemungkinan ini bisa terjadi bila ternyata jawaban soal yang akan digunakan dalam TTS tidak dapat disusun.

B. Saran

Saran yang diberikan untuk pengembangan sistem selanjutnya adalah :

1. Penerapan algoritma genetika hendaknya dapat dikembangkan untuk membuat TTS angka (*figjig*).
2. Penerapan algoritma genetika hendaknya dapat dikembangkan untuk membuat TTS berbasis multimedia.

DAFTAR PUSTAKA

- Desiani, A. & Arhami, M, 2006, *Konsep Kecerdasan Buatan*. Yogyakarta: Andi.
- Dewobroto, Wiryanto, 2003, *Aplikasi SAIN dan TEKNIK dengan VISUAL BASIC 6.0*. Jakarta: PT. Elex Media Komputindo.
- Kusumadewi, S, 2003, *Artificial Intelligence*. Yogyakarta: Graha Ilmu.
- Kuswadi, S, 2007, *Kendali Cerdas: Teori dan Aplikasi Praktisnya*. Yogyakarta: Andi.
- Ladjamuddin B, Al-Bahra, 2006, *Rekayasa Perangkat Lunak*. Yogyakarta: Graha Ilmu.
- Munir, R, 2004, *Pengolahan Citra Digital dengan Pendekatan Algoritma*. Bandung: Informatika.
- Pandjaitan, L. W, 2007, *Dasar-Dasar Komputasi Cerdas*. Yogyakarta: Andi.
- Simarmata, J. & Chandra, T, 2007, *Grafika Komputer*. Yogyakarta: Andi.
- Sutoyo, T. et.al, 2009, *Teori Pengolahan Citra Digital*. Yogyakarta: Andi.
- Yuswanto, 2002, *Visual BASIC 6.0: Pemrograman Grafis & Multimedia*. Surabaya: Prestasi Pustaka Publisher.
- Wikipedia, 2010, Teka-Teki Silang, Jurnal (Online) http://id.wikipedia.org/wiki/Teka-teki_silang, diakses 10 April 2010