

Analisis Penerapan Algoritma *Backtracking* Dalam Pencarian Solusi Game “Crossword Puzzle”

Widya Wardani¹, Dian Intania Savitri H², Allentine Tanujaya³

*Laboratorium Ilmu dan Rekayasa Komputasi
Departemen Teknik Informatika, Institut Teknologi Bandung
Jl. Ganesha 10, Bandung*

E-mail : if13076@students.if.itb.ac.id¹, if13081@students.if.itb.ac.id²,
if13104@students.if.itb.ac.id³

ABSTRAK

Game komputer merupakan aplikasi *software* yang lazim dimanfaatkan oleh *user*. Dengan jenis yang bermacam-macam dan tampilan yang menarik, *game* komputer termasuk *software* yang diminati oleh berbagai kalangan. Selain karena tampilan dan aplikasinya relatif menarik, *game* komputer juga disinyalir dapat menjadi salah satu sarana *refreshing* yang cukup menyenangkan terutama bagi orang yang telah terbiasa menggunakan komputer. Pada makalah ini, penulis mencoba membahas tentang *game* “*Crossword Puzzle*” dan menganalisis penerapan algoritma *backtracking* sebagai salah satu cara penyelesaian persoalan pada *game* ini.

Berbagai data dalam makalah ini penulis peroleh dari berbagai sumber yang berkaitan. Sumber utama yang menjadi referensi penulis adalah Diktat Kuliah IF2251 “Strategi Algoritmik” yang disusun oleh Ir. Rinaldi Munir, M.T. Selain dari sumber tersebut kami juga mendapatkan referensi dari situs-situs yang berkaitan dengan penerapan algoritma yang telah kami pelajari untuk aplikasi *game* “*Crossword Puzzle*” di mana komputer diminta untuk memasukkan kata yang berkesesuaian dengan kotak yang tersedia.

Melalui pembahasan makalah ini, penulis menemukan bahwa *game* “*Crossword Puzzle*” menggunakan algoritma *backtracking* (runut balik) di mana komputer akan memasukkan kata yang panjangnya sama dengan kotak yang tersedia sampai seluruh kotak terisi. Jika dalam proses pengisian ternyata terjadi ketidaksinambungan maka akan dilakukan *backtracking* (runut balik).

Kata Kunci : *backtracking*, *game*, crossword puzzle

1. PENDAHULUAN

Semenjak dahulu, permainan (*game*) merupakan hal yang sangat menarik bagi sebagian besar masyarakat dunia. Dengan berkembangnya berbagai bentuk *game*, baik berupa *game* yang masih sederhana maupun yang sudah menggunakan teknologi yang canggih, *game* menjadi sangat populer di berbagai tempat.

Salah satu bentuk *game* yang dewasa ini sudah tidak asing adalah *game* yang dimainkan di komputer. *Game* merupakan aplikasi yang cukup diminati karena bisa dimanfaatkan sebagai ajang *refreshing*

dan olah otak. Bahkan tidak jarang, aplikasi *game* pada komputer menimbulkan kecanduan bagi *user*.

Game “*Crossword Puzzle*” pada dasarnya merupakan suatu aplikasi *game* yang memungkinkan user menginput kata yang diambil dari kumpulan kata atau *database* yang tersedia dan mengisikannya secara tepat pada kotak yang telah tersedia. Namun di sini, analisis yang penulis lakukan adalah berkenaan dengan pengisian kata-kata ke dalam kotak yang telah tersedia oleh komputer sebagai player. Dalam hal ini, algoritma yang dipergunakan adalah algoritma *backtracking* (runut balik) yang menangani ketidaksinambungan yang dapat terjadi jika memasukkan kata-kata yang salah.

2. PENCARIAN SOLUSI GAME : “CROSSWORD PUZZLE”

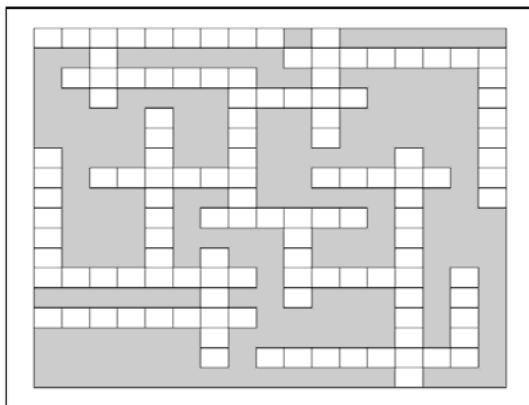
2.1. Deskripsi Umum *Game crossword puzzle*

Crossword puzzle merupakan suatu *game* yang templatnya berbentuk segi empat yang terdiri dari kumpulan kotak-kotak berwarna hitam dan putih dan 2 lajur yaitu mendatar (kumpulan kotak yang membentuk satu baris dan beberapa kolom) dan menurun (kumpulan kotak yang membentuk satu kolom dan beberapa baris). Untuk menyelesaikan permainan ini, keseluruhan kotak yang berwarna putih harus terisi dengan kata-kata yang tersedia dalam kumpulan kata yang ada.

Secara spesifik *Crossword Puzzle* merupakan suatu *game* yang memungkinkan user memasukkan kata yang bersesuaian dengan panjang kotak yang tersedia secara berkesinambungan sampai seluruh kotak terisi penuh. Aturan pengisian kata-kata tersebut berhubungan dengan penyamaan jumlah kotak dengan jumlah karakter pada kata dan pengisian kata-kata ke dalam kotak pada *crosseord puzzle* secara berkesinambungan.

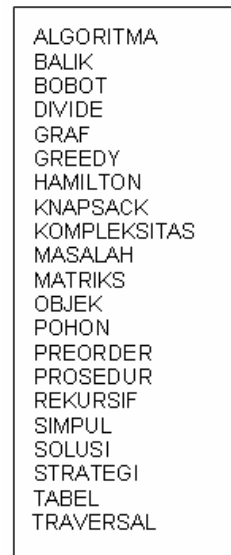
Dalam makalah ini, penulis mengasumsikan yang berfungsi sebagai player adalah komputer itu sendiri. Untuk *game* ini, telah tersedia template berisi kotak kosong yang akan menerima input kata yang panjangnya sesuai dengan jumlah karakter (kotak) yang tersedia.

Contoh template kotak yang bisa diinput huruf



Gambar1. Template Crossword Puzzle

Contoh tampilan kumpulan kata yang harus diinput :



Gambar2. Tampilan kumpulan kata

2.2. Deskripsi Umum Algoritma *Backtracking*

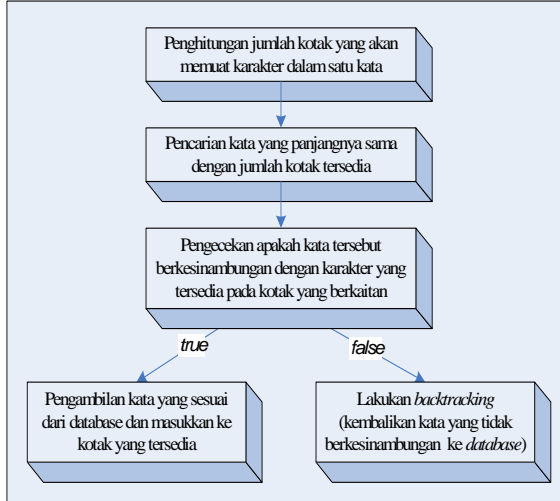
Algoritma *backtracking* pertama kali diperkenalkan oleh D.H. Lehmer pada tahun 1950. Dalam perkembangannya beberapa ahli seperti RJ Walker, Golomb, dan Baumert menyajikan uraian umum tentang *backtracking* dan penerapannya dalam berbagai persoalan dan aplikasi.

Algoritma *backtracking* (runut balik) merupakan salah satu metode pemecahan masalah yang termasuk dalam strategi yang berbasis pencarian pada ruang status. Algoritma *backtracking* bekerja secara rekursif dan melakukan pencarian solusi persoalan secara sistematis pada semua kemungkinan solusi yang ada. Oleh karena algoritma ini berbasis pada algoritma *Depth-First Search (DFS)*, maka pencarian solusi dilakukan dengan menelusuri suatu struktur berbentuk pohon berakar secara preorder. Proses ini dicirikan dengan ekspansi simpul terdalam lebih dahulu sampai tidak ditemukan lagi suksesor dari suatu simpul.

2.3. Algoritma *Backtracking* Sebagai Pemecahan Solusi

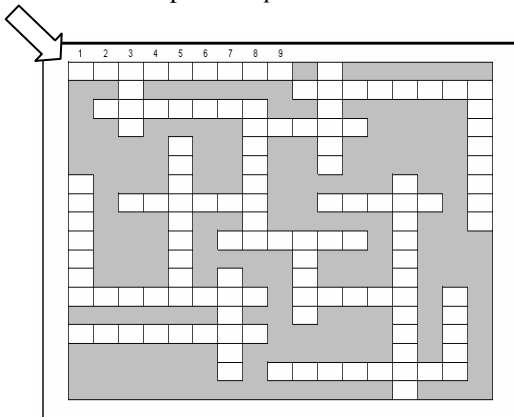
Penggunaan algoritma *backtracking* ini akan terlihat dalam proses pengisian kotak di mana terdapat beberapa kemungkinan kata yang sesuai. Pada pengisian selanjutnya, karakter yang dipakai untuk kata lain (terlihat pada kotak yang tersambung baik secara mendatar maupun menurun) akan dicocokkan dengan kumpulan kata yang tersedia. Jika tidak ditemukan kata yang berkesesuaian maka kata yang berkaitan dengannya akan dikembalikan ke kumpulan kata tadi (*database* yang memuat kumpulan kata).

Metode pengisian kata ke dalam kotak-kotak yang tersedia dapat dilakukan secara rekursif sedangkan proses pencarian kata pada *database* dilakukan secara iteratif. Proses pencarian solusi ini terdiri atas beberapa tahap yaitu:

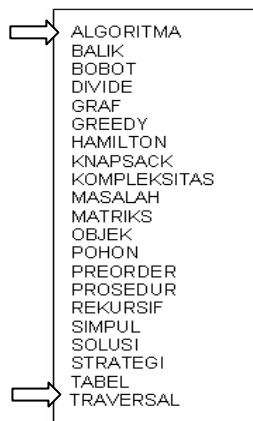


Contoh pencarian solusi yang dapat dilakukan sebagai salah satu penerapan algoritma *backtracking*

- a. Perhitungan jumlah kotak yang pertama kali ditemukan pada *template*.

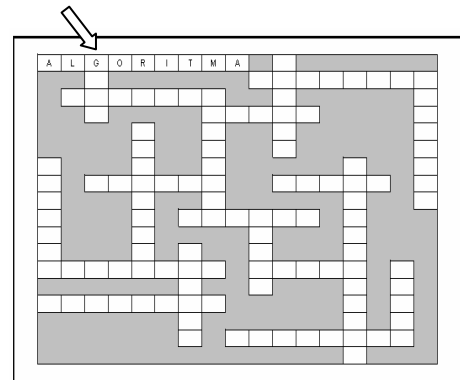


- b. Pencarian kata yang panjangnya sama dengan jumlah kotak yang telah dihitung

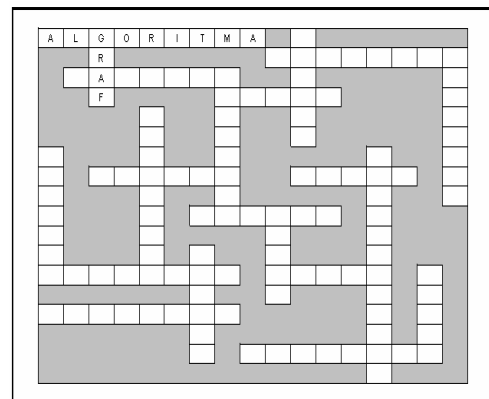


Ket :

ada 2 kata yang panjang karakternya sama-sama 9 (algoritma dan traversal). Ketika pencarian dilakukan, kata yang panjangnya cocok yang ditemukan pertama kali akan dimasukkan ke dalam *cross word puzzle*. Pada kasus ini, 'algoritma' adalah kata yang pertama kali dimasukkan sehingga tidak perlu adanya pencocokan karakter. Pencocokan karakter akan dilakukan untuk mengisi kolom 3 secara menurun (lanjutan huruf 'g')



- c. Pengecekan kata
Setelah dilakukan pencarian kata di *database* yang memenuhi spesifikasi terdiri dari 4 huruf dan diawali dengan huruf 'g' maka kata yang didapat adalah 'graf' sehingga *crossword puzzle* nya menjadi :



- d. Jika Pengecekan kata menghasilkan *true*, maka dilakukan proses pengisian *crossword* secara rekursif dan proses kembali ke tahap awal. Sehingga didapatkan hasil akhir menjadi seperti gambar di bawah ini :

