

MERANCANG DAN MEMBUAT PROGRAM PERMAINAN QUAD DENGAN METODE FORWARD-CHAINING



Oleh :

Nama : ZAINUDDIN
NIM : 91.1047
NIRM : 91.7.085.31132.00531
Program : S1 (Strata Satu)
Jurusan : Manajemen Informatika

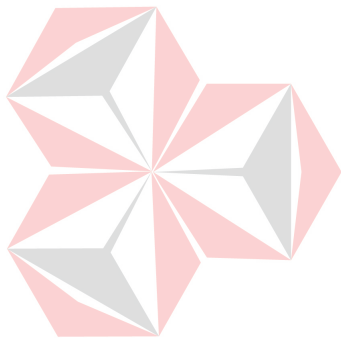
SEKOLAH TINGGI
MANAJEMEN INFORMATIKA & TEKNIK KOMPUTER
SURABAYA

1997

MERANCANG DAN MEMBUAT PROGRAM PERMAINAN QUAD DENGAN METODE FORWARD-CHAINING

SKRIPSI

**Diajukan sebagai salah satu syarat untuk menyelesaikan
Program Sarjana Komputer**



UNIVERSITAS
Dinamika

Oleh :

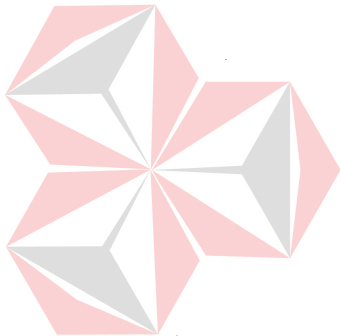
**Nama : ZAINUDDIN
NIM : 91.1047
NIRM : 91.7.085.31132.00531
Program : S1 (Strata Satu)
Jurusan : Manajemen Informatika**

**SEKOLAH TINGGI
MANAJEMEN INFORMATIKA & TEKNIK KOMPUTER
SURABAYA**

1997

**MERANCANG DAN MEMBUAT
PROGRAM PERMAINAN QUAD
DENGAN METODE FORWARD-CHAINING**

Telah diperiksa, diuji dan disetujui



UNIVERSITAS
Dinamika

Surabaya, 24 Juni 1997

Mengetahui

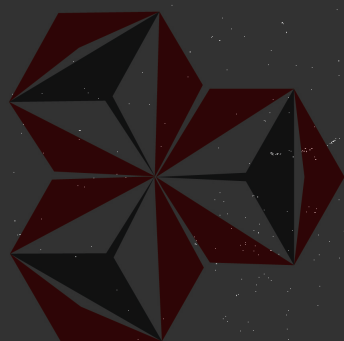
A circular stamp is visible, partially obscured by a handwritten signature. The signature is written in black ink and appears to be 'R. Susilo'. The stamp contains text that is mostly illegible but seems to be related to the university's administrative process.

Ir. Ronny S. Susilo, MM
Pembantu Ketua I

Menyetujui :

A handwritten signature in black ink, which appears to be 'P. Agus D.', is written over a horizontal line.

Ir. Purwadi Agus D., MSc
Pembimbing



UNIVERSITAS
Dinamika

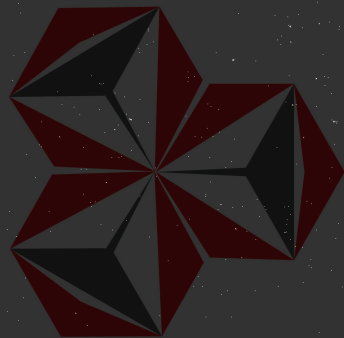
ABSTRAKSI

Perkembangan komputer sekarang ini telah membuat segala pekerjaan beralih ke komputer, salah satu contohnya adalah permainan. Permainan komputer sekarang ini sangat interaktif, mulai dari permainan arcade sampai ke permainan untuk mengasah otak.

Untuk mengetahui suatu masalah pada permainan yang menggunakan kemampuan berpikir, ketelitian dan kecerdasan seseorang dapat dilihat dari ide yang sederhana tetapi memiliki teknik dan metode pembuatan yang cukup tinggi. Seperti pada permainan Go-moku yaitu permainan sederhana yang menggunakan sebuah papan permainan dan alat tulis dimana pemain akan berusaha saling menjatuhkan / menghalangi untuk membentuk sebuah garis lurus dan siapa yang berhasil pertama kali membuatnya maka dinyatakan sebagai pemenangnya.

Tujuan tugas akhir ini adalah merancang dan membuat sebuah program permainan Quad yang merupakan kembangan dari permainan Go-moku yang selama ini permainan ini masih dimainkan secara manual.

Dalam program permainan Quad terdapat 2 jenis pemain yaitu manusia dan komputer. Pemain (manusia) pada umumnya mempunyai strategi sendiri-sendiri untuk menentukan posisi mana pada papan permainan yang akan ditempati oleh buah yang dipegangnya. Sedangkan untuk pemain lain (komputer) diperlukan suatu metode yang baik, efektif dan dapat membaca permainan lawan dengan berdasarkan pada metode Forward-Chaining dan teknik Depth-First.



UNIVERSITAS
Dinamika

KATA PENGANTAR

Puji syukur kehadirat Allah SWT dengan rahmat serta ridlo-Nya, Alhamdulillah penyusunan laporan tugas akhir ini dapat terselesaikan.

Laporan tugas akhir ini disusun sebagai salah satu persyaratan untuk menyelesaikan program sarjana di Sekolah Tinggi Manajemen dan Teknik Komputer Surabaya.

Dalam menyusun laporan penulis banyak mendapat bimbingan, petunjuk serta bantuan dari berbagai pihak, baik secara moril dan materiil yang tidak ternilai harganya. Untuk itu dalam kesempatan ini, penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada :

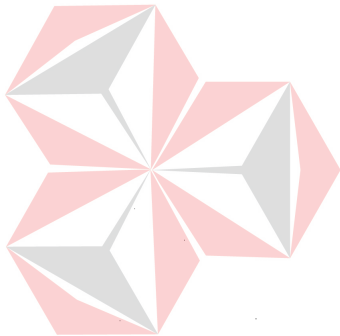
1. Bapak Ir. Purwadi Agus Darwito, MSc. yang telah mengorbankan waktu dan memberikan bimbingan kepada penulis, sehingga laporan tugas akhir ini dapat terselesaikan.
2. Bapak Haryanto, Skom. yang telah memberikan ijin kepada penulis untuk mengerjakan tugas akhir ini.
3. Seluruh staff dan karyawan STIKOM yang telah banyak membantu.
4. Kedua orang tua, saudara penulis yang senantiasa memberikan dorongan serta doanya untuk menyelesaikan tugas akhir ini.
5. Teman-teman angkatan '91 yang banyak membantu dalam penulisan maupun dalam hal tukar pendapat tentang teknik pemrograman.
6. Saudara anggota STI STIKOM Surabaya yang memberikan dorongan dalam hal doa.

Penulis menyadari bahwa dalam penyusunan tugas akhir ini masih jauh dari harapan. Oleh karena itu saran dan kritik yang positif akan sangat membantu penulis dalam menyusun laporan-laporan berikutnya.

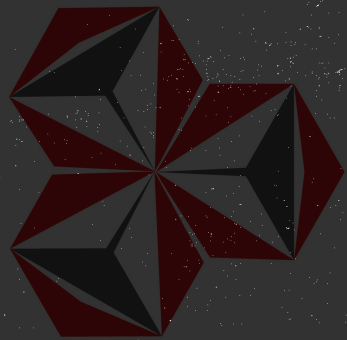
Akhir kata penulis berharap semoga tugas akhir ini akan bermanfaat bagi pembaca untuk mengisi waktu luang.

Surabaya, 24 Juni 1997

Penulis



UNIVERSITAS
Dinamika

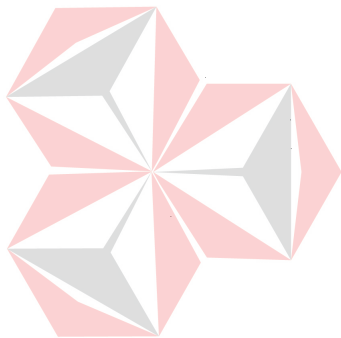


UNIVERSITAS
Dinamika

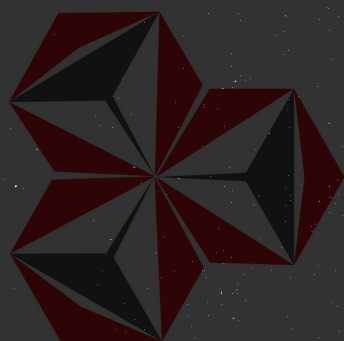
DAFTAR ISI

	Halaman
ABSTRAKSI.....	i
KATA PENGANTAR	ii
DAFTAR ISI.....	iv
DAFTAR TABEL.....	vi
DAFTAR GAMBAR.....	vii
DAFTAR LAMPIRAN.....	viii
BAB I. PENDAHULUAN.....	1
1.1. Latar Belakang Masalah.....	1
1.2. Tujuan	2
1.3. Batasan Permasalahan.....	2
1.4. Metode Pemecahan Masalah.....	3
1.5. Sistematika Penyusunan Tugas Akhir.....	3
BAB II. LANDASAN TEORI.....	5
2.1. Aturan Umum Permainan Quad.....	5
2.2. Metode Forward-Chaining.....	6
2.3. Teknik Pelacakan Depth-First.....	8
2.4. Bahasa Pemrograman Delphi.....	10
BAB III. PERMASALAHAN.....	15
BAB IV. IMPLEMENTASI DAN PEMBAHASAN	17
4.1. Perangkat Pendukung.....	17
4.2. Rancangan Perangkat Lunak.....	17

	Halaman
4.3. Pengujian.....	30
4.4. Analisa Hasil Pengujian.....	40
KESIMPULAN DAN SARAN.....	41
DAFTAR PUSTAKA	42
LAMPIRAN.....	43



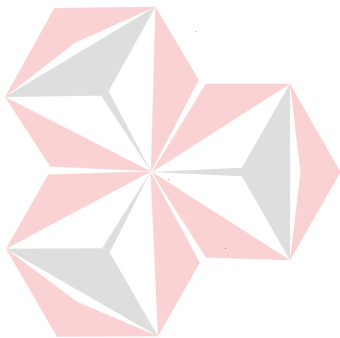
UNIVERSITAS
Dinamika



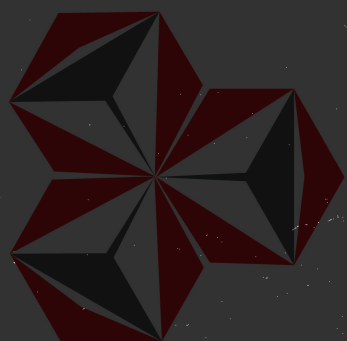
UNIVERSITAS
Dinamika

DAFTAR TABEL

	Halaman
Tabel 4.1. Percobaan 1 Pengujian strategi komputer.....	33
Tabel 4.2. Percobaan 2 Pengujian strategi komputer.....	34
Tabel 4.3. Percobaan 3 Pengujian strategi komputer.....	36
Tabel 4.4. Percobaan 4 Pengujian strategi komputer.....	38
Tabel 4.5. Percobaan 5 Pengujian strategi komputer.....	39



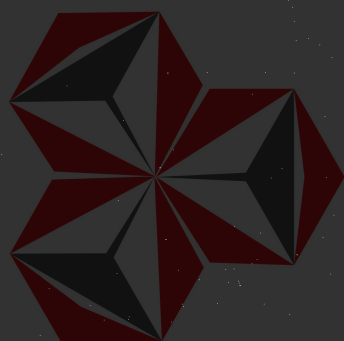
UNIVERSITAS
Dinamika



UNIVERSITAS
Dinamika

DAFTAR GAMBAR

	Halaman
Gambar 2.1. Langkah permainan Quad	6
Gambar 2.2. Metode Forward-Chaining	7
Gambar 2.3. Metode Backward-Chaining	8
Gambar 2.4. Teknik Depth-First Search	9
Gambar 2.5. Teknik Breadth-First Search	10
Gambar 4.1. Diagram alir program utama permainan Quad.....	19
Gambar 4.2. Diagram alir mode manual.....	20
Gambar 4.3. Diagram alir mode komputer	21
Gambar 4.4. Kemungkinan merah menang.....	32
Gambar 4.5. Percobaan 1	34
Gambar 4.6. Percobaan 2	35
Gambar 4.7. Percobaan 3	37
Gambar 4.8. Percobaan 4	38
Gambar 4.9. Percobaan 5	40



UNIVERSITAS
Dinamika

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Prg_Utama.pas.....	43
Lampiran 2. Ta_Quad.dpr.....	61
Lampiran 3. Prg_Depan.pas.....	62
Lampiran 4. Prg_Info.pas.....	63
Lampiran 5. Prg_Menang.pas.....	65
Lampiran 6. Prg_Musik.pas.....	66
Lampiran 7. Prg_Petunjuk.pas.....	67



UNIVERSITAS
Dinamika

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Saat ini banyak sekali jenis permainan yang ada di masyarakat, mulai dari permainan klasik / sederhana misal othelo, go-moku / tic-tac-toe sampai ke permainan modern seperti nintendo ataupun permainan simulasi yang menggunakan kemuktahiran teknologi. Tujuan dari permainan selain berfungsi sebagai sarana hiburan juga sebagai sarana pendidikan, misalnya untuk melatih kecerdasan, ketelitian dan menambah pengetahuan seseorang.

Ada banyak permainan yang berfungsi melatih ketelitian dan kecerdasan seseorang, salah satu contoh permainan tersebut adalah Quad, yaitu suatu permainan yang diciptakan oleh seorang mahasiswa pada sebuah universitas di Eropa dimana permainan ini merupakan perkembangan / modifikasi dari permainan go-moku. Inti dari permainan Quad ini adalah bagaimana pemain berusaha untuk membentuk suatu bujursangkar pada sebuah papan permainan dengan meletakkan empat buah titik sebagai landasannya.

Dengan semakin banyaknya pekerjaan-pekerjaan dan sistem yang telah terkomputerisasi, maka pekerjaan-pekerjaan tersebut akan lebih interaktif, lebih mandiri, dan lebih mudah untuk menggunakannya. Di antaranya adalah permainan, sekarang ini banyak sekali permainan yang biasanya untuk memainkannya saja diperlukan suatu peralatan yang banyak dan ada pula yang berat, sehingga sulit untuk dipindahkan. Karena itu, sekarang ini banyak permainan dialihkan menjadi suatu

permainan dalam komputer. Dengan latar belakang permasalahan di atas, permainan Quad akan diimplementasikan ke dalam program permainan komputer.

1.2. Tujuan.

Tujuan pembuatan tugas akhir ini adalah membuat suatu program permainan Quad dengan menggunakan metode *Forward-Chaining* dan teknik pelacakan *Depth-First*. Permainan Quad ini diharapkan dapat digunakan sebagai sarana hiburan dan sarana pendidikan.

1.3. Batasan Permasalahan.

Batasan permasalahan yang penulis gunakan dalam tugas akhir ini adalah sebagai berikut :

- a. Program permainan Quad ini hanya dapat digunakan oleh 2 (dua) orang pemain atau 1 (satu) orang pemain dan sebagai lawannya adalah komputer.
- b. Metode yang digunakan untuk menentukan langkah-langkah komputer adalah metode *Forward-Chaining*.
- c. Teknik pencarian / pelacakan komputer dengan teknik *Depth-First*.
- d. Memakai papan permainan (board) dalam bentuk bujursangkar dengan 11 x 11 lingkaran didalamnya dan setiap sudutnya lingkaran tersebut dihapus / dihilangkan, menjadi total semua lingkaran adalah 117 buah.
- e. Program permainan ini digunakan pada komputer *PC* yang stand-alone / berdiri sendiri dan sistem operasinya berbasis windows 32-bit.
- f. Dalam pembuatan program permainan ini, penulis memilih bahasa pemrograman *Borland Delphi 2.0* dengan memakai sistem operasi *Microsoft Windows 95*.

1.4. Metode Pemecahan Masalah.

Untuk mencapai tujuan di atas, maka penulis melakukan studi eksperimen untuk mengetahui secara jelas aturan permainan Quad selain dapat informasi singkat dari penciptanya dan studi pustaka dengan prosedur sebagai berikut :

- a. Mempelajari aturan permainan Quad yang dilakukan secara manual.
- b. Studi pustaka tentang metode, teknik dan strategi yang akan digunakan.
- c. Menentukan bahasa pemrograman.
- d. Pembuatan program permainan.
- e. Pengujian program.
- f. Penyusunan laporan tugas akhir.

1.5. Sistematika Penyusunan Tugas Akhir.

Dalam penyusunan laporan tugas akhir ini dibagi menjadi beberapa bab yang disusun sebagai berikut :

BAB I. PENDAHULUAN

Menjelaskan tentang latar belakang permasalahan yang mendorong penulis untuk menggunakan permainan Quad sebagai permasalahan dalam tugas akhir, tujuan yang akan dicapai dan batasan permasalahan.

BAB II. LANDASAN TEORI

Menjelaskan aturan dan teori yang digunakan penulis untuk mendukung penyelesaian pembuatan program permainan Quad. Teori yang digunakan adalah metode pengambilan keputusan *Forward-Chaining*, teknik

pelacakan *Depth-First*, aturan umum permainan Quad dan pemrograman *Borland Delphi*.

BAB III. PERMASALAHAN

Menjabarkan permasalahan yang ditemukan dalam proses pembuatan program permainan Quad. Di antaranya adalah bagaimana membentuk program bersifat *user friendly*, pengendalian permainan agar sesuai dengan aturan yang berlaku dan penerapan metode pengambilan keputusan *Forward-Chaining*, teknik pelacakan *Depth-First*.

BAB IV. IMPLEMENTASI DAN PEMBAHASAN

Menjelaskan perencanaan pembentukan program. Algoritma program permainan Quad yang dibuat beserta pembahasan dan proses analisa program.

BAB V. KESIMPULAN DAN SARAN

Berisi kesimpulan akhir yang didapat berdasarkan pembahasan dan analisa program serta saran yang diperlukan untuk pengembangan program permainan Quad.

BAB II

LANDASAN TEORI

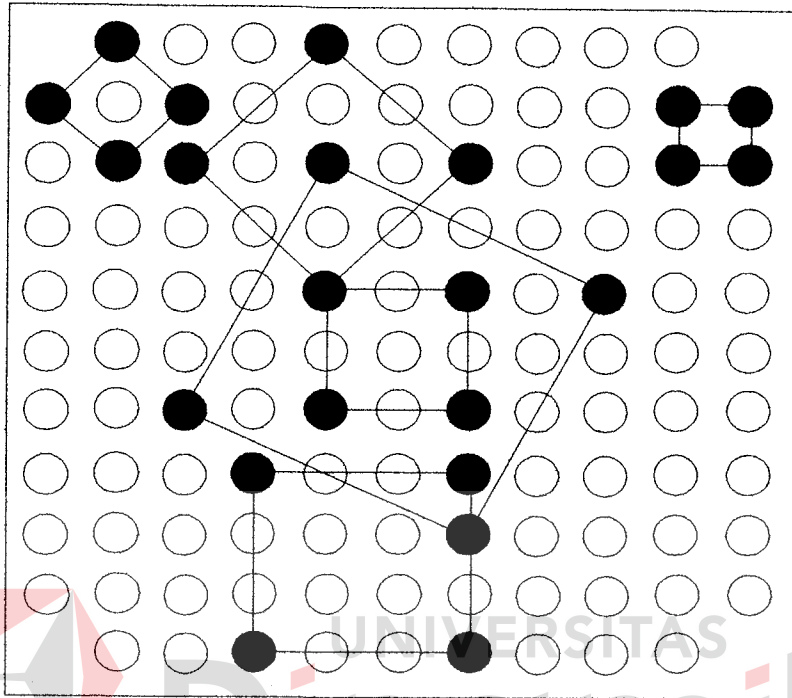
Di dalam setiap pembuatan program permainan terdapat aturan umum yang berlaku untuk permainan itu sendiri dan teori-teori yang digunakan untuk mewujudkan aturan tersebut.

2.1. Aturan Umum Permainan Quad.

Sesuai dengan penjelasan secara singkat dari pencipta permainan ini maka peraturan permainan Quad dijelaskan sebagai berikut :

- a. Jumlah pemain hanya 2 (dua) orang pemain.
- b. Setiap pemain mendapatkan 20 buah merah / biru sebagai buah penyerang dan 6 buah buah putih sebagai buah blok.
- c. Buah merah jalan terlebih dahulu, baru buah biru atau blok putih milik biru.
- d. Papan permainan yang digunakan adalah papan permainan berbentuk bujursangkar yang berisi 11 x 11 lingkaran, pada setiap sudutnya lingkaran tersebut dihilangkan sehingga jumlah keseluruhannya ada 117 buah.
- e. Permainan dapat dikatakan menang apabila salah satu pemain dapat membuat / meletakkan 4 buah titik (merah / biru) yang membentuk bujursangkar (setiap sisinya sama) pada papan permainan tersebut. Sedangkan permainan dikatakan seri apabila buah penyerang dan buah bloknya habis dan belum membentuk 4 buah titik. Apabila buah penyerang habis dan buah blok masih ada, maka buah

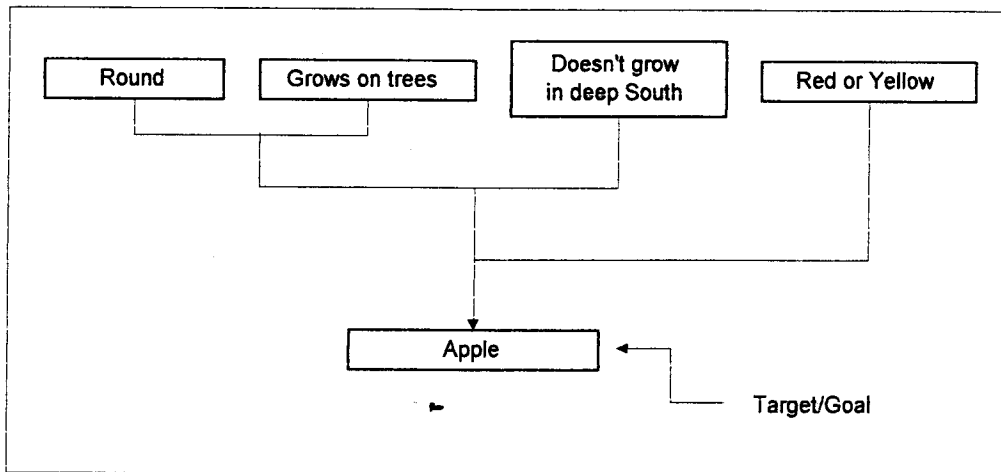
blok yang paling banyak disebut pemenangnya. Contoh langkah permainan Quad seperti pada gambar dibawah ini :



Gambar 2.1. Langkah permainan Quad

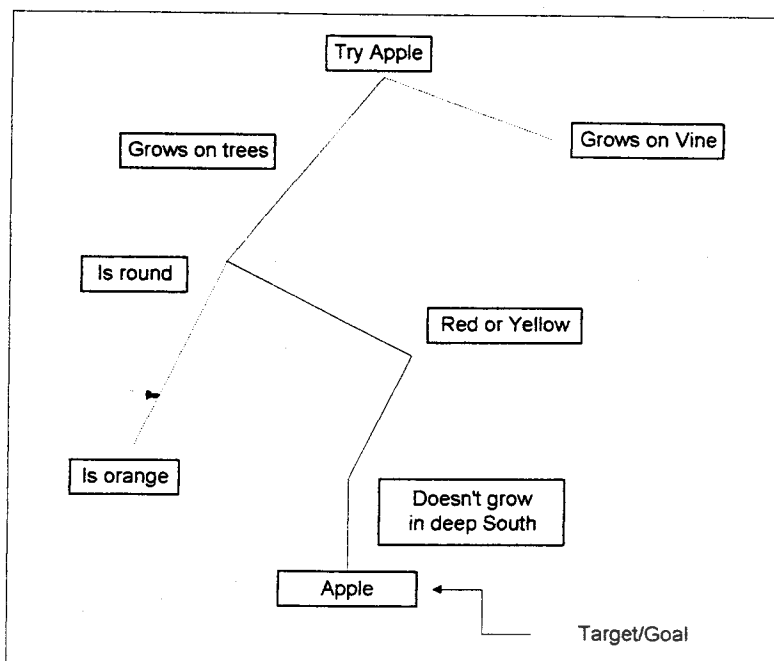
2.2. Metode Forward-Chaining.

Secara umum seperti yang ditulis dalam buku *Artificial Intelligence Using C The C Programmer's Guide to AI Techniques* oleh Herbert Schildt, *Forward-Chaining* adalah salah satu metode yang memulai langkah pertama dan berikutnya dari informasi / data sebelumnya untuk mencari suatu solusi yang sesuai dengan aturan yang ada disebut juga *data-driven*. Seperti gambar dibawah ini untuk mendefinisikan buah apel berdasarkan pada metode *Forward-Chaining* :



Gambar 2.2. Metode *Forward-Chaining*

Sedangkan Metode *Backward-Chaining* yang disebut juga *object-driven* yaitu memulai langkah pertama dan berikutnya dengan menentukan sebuah hipotesa awal yang akan diuji berdasarkan informasi yang akan diterima. Apabila informasi yang diterima benar maka hipotesa awal tersebut adalah solusinya, apabila tidak benar maka hipotesa akan diganti oleh hipotesa berikutnya dan seterusnya. Misalnya seperti pada gambar dibawah ini :

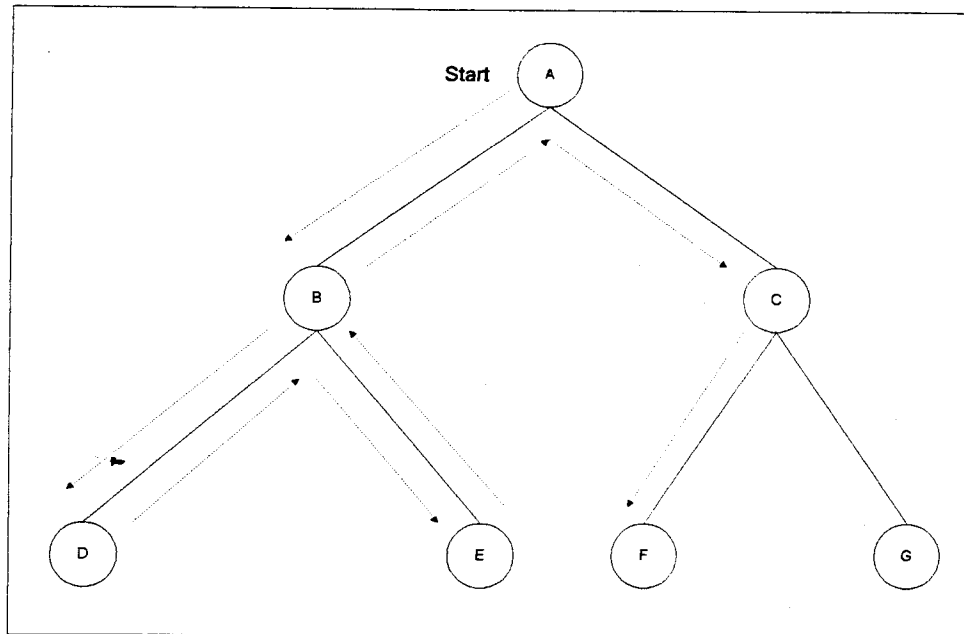


Gambar 2.3. Metode *Backward-Chaining*

Sesuai dengan penjelasan diatas tentang perbedaan metode *Forward-Chaining* dengan metode *Backward-Chaining*, maka dipilih metode *Forward-Chaining* sebagai strategi untuk komputer. Karena dalam permainan Quad langkah yang akan dilakukan berikutnya berdasarkan langkah sebelumnya. Hal ini sesuai dengan prinsip dari metode *Forward-Chaining*.

2.3. Teknik Pelacakan Depth-First.

Teknik ini digunakan bukan untuk mencari kecepatan pencarian tetapi untuk mendapatkan ketelitian yang sangat tinggi dan secara bertahap sesuai dengan tingkatannya. Sebagai ilustrasinya terdapat pada gambar dibawah ini :

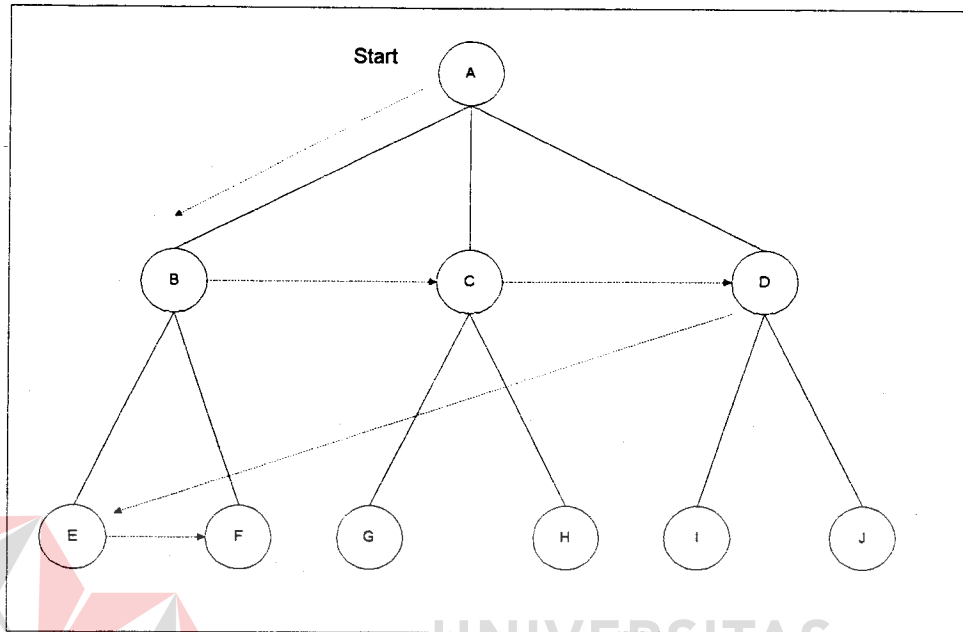


Gambar 2.4. Teknik *Depth-First Search*

Depth-First akan mencari tujuan / goal, misalnya F adalah goal yang dicari, dimana melalui gambar di atas akan dicari secara berurutan mulai dari A-B-C-D-E-B-A-C-F. Di awali dari titik A tingkatan pertama akan menuju ke kanan kemudian menuju ke tingkatan berikutnya, (B) akan mencari apakah ada tingkatan berikutnya atau ada berapa cabang. Apabila cabang kanan sudah tidak ada, maka akan kembali menuju ke (B) lagi untuk mencari cabang kiri ada atau tidak dan seterusnya. Dengan demikian target / goal (F) dapat ditemukan.

Sedangkan teknik pelacakan *Breath-First* melakukan pelacakan target / goal dari simpul pada tingkatan awal apakah memiliki akar atau tidak, apabila memiliki beberapa akar maka akan dicari targetnya dari akar paling kiri menuju ke akar kanan. Bila tidak memiliki akar maka akan dilanjutkan menuju ke tingkatan berikutnya,

sampai target / goal didapatkan. Untuk lebih jelasnya dapat melihat gambar dibawah ini :



Gambar 2.5. Teknik *Breadth-First Search*

Dalam program permainan ini dipilih teknik pelacakan *Depth-First* karena untuk mencari posisi pemain komputer lebih teliti dan di dalam permainan ini mudah di implementasikan dengan memakai teknik tersebut. Komputer akan membaca papan permainan dari baris pertama sampai baris terakhir untuk mencari letak posisi lawan dan disimpan sebagai acuan untuk menjalankan langkah berikutnya sesuai dengan metode *Forward-Chaining* yang telah dijelaskan di atas.

2.4. Bahasa Pemrograman Delphi.

Bahasa pemrograman yang dipakai *Delphi* berdasarkan bahasa pemrograman *Pascal*, yang berarti semua keunggulan yang dimiliki *Pascal* juga milik *Delphi*.

Delphi sepenuhnya menerapkan metode pemrograman *Object Oriented Programming*, suatu metode pemrograman yang mampu menangani program dalam ukuran besar karena memiliki sistem organisasi yang baik. *Delphi* bekerja dalam lingkungan sistem operasi *windows* dan mudah dalam menyusun program aplikasi karena memakai komponen-komponen yang menghemat penulisan program. Dalam membuat *window*, kotak dialog maupun perangkat kontrol lainnya seperti tombol perintah (*command button*), menu *pull-down*, kotak combo (*combo box*) dan sebagainya telah disediakan oleh *Delphi* (tidak perlu menuliskan kode yang rumit untuk membuatnya).

Selain itu keunggulan *Delphi* adalah dalam hal kecepatan proses kompilasi program, file yang sudah dikompilasi oleh *Delphi* menghasilkan satu file saja dan filenya relatif kecil serta dapat diakses oleh sistem operasi yang sama pada tempat yang berbeda.

Delphi membagi program menjadi 2 bagian utama yaitu bagian primer dan bagian sekunder. Bagian primer berupa *file* program yang mengkoordinasi seluruh program, *file* ini dicirikan dari ekstensinya yaitu : ***.dpr**. Sedangkan bagian sekunder merupakan bagian yang melaksanakan tugas-tugas tertentu dan lazim disebut *unit file*. Bagian ini dicirikan oleh ekstensi ***.pas**. Pada waktu menyusun program *Delphi*, yang ditulis hanya kode untuk *file-file unit* (*.pas), sedangkan *file project* akan otomatis terbentuk.

2.4.1. Program Primer.

Secara umum perintah program primer *Delphi* adalah sebagai berikut :

1. **program** Project1;
2. **uses**
3. *Form,*
4. *Or1 in 'ORI.PAS' (Form1);*
5. *{SR *.RES}*
6. **begin**
7. *Application.CreateForm(TForm1, Form1);*
8. *Application.Run;*
9. **end.**

2.4.2. Program Sekunder.

Secara umum perintah program sekunder Delphi adalah sebagai berikut :

1. **unit** Bahasa1;
2. **interface**
3. **Uses**
4. *SysUtils, WinTypes, WinProcs, Message, Classes, Graphics,*
5. *Controls, Forms, Dialogs, StdCtrls, ColorGrd;*
- 6.
7. **type**
8. *Tform1=class(Tform)*
9. *ColorGrid1:TcolorGrid;*
10. *Button1:Tbutton;*
11. *Edit1:Tedit;*
12. **Procedure** Button1Click(Sender:Tobject);
13. **Procedure** ColorGrid1Change(Sender:Tobject);
14. **Private**
15. *{private declarations}*
16. **public**
17. *{public declarations}*
18. **end;**
- 19.
20. **var**
21. *form1:Tform;*
22. **implementation**
23. *{SR *.DFM}*
- 24.
25. **procedure** Tform1.Button1Click(Sender:Tobject);
26. **begin**
27. *Close;*
28. **End;**
- 29.
30. **procedure** Tform1.ColorGrid1Change(Sender:Tobject);
31. **begin**
32. *edit1.Color:=ColorGrid1.ForegroundColor;*
33. **end;**
- 34.
35. **end.**

2.4.3. Integrated Development Environment (IDE).

IDE adalah area kerja, tempat menghasilkan program aplikasi. Delphi melayani pemakainya dengan fasilitas yang lengkap, diantaranya dengan menyediakan *IDE*. Dengan *IDE* pembuat program tidak perlu repot dengan melakukan akses fungsi-fungsi yang diperlukan, karena seluruh kebutuhan sudah disediakan dalam satu tampilan. Misalnya, untuk menuliskan kode program, pembuat program tinggal menuliskannya pada *Code Editor* yang telah terbuka dan secara otomatis aktif (siap pakai) bila diperlukan.

Demikian pula untuk keperluan lain seperti *debugging* (pencarian kesalahan program), *compiling* (penerjemahan program) maupun *running* program. Bahkan untuk pembuatan file eksekusi (file yang dapat berjalan sendiri, yang dicirikan dengan nama ekstensi *.exe) hanya memerlukan satu langkah, yaitu dengan perintah *Compile*.

2.3.1. Rumus Forward-Chaining dan Depth-First.

Rumus yang digunakan pada program permainan Quad berdasarkan pada aturan main (*rule base*) untuk membentuk bujursangkar adalah sebagai berikut :

$$KolomDiff = Kolom2 - Kolom1 \quad \dots\dots\dots (persamaan 2.1)$$

$$BarisDiff = Baris2 - Baris1 \quad \dots\dots\dots (persamaan 2. 2)$$

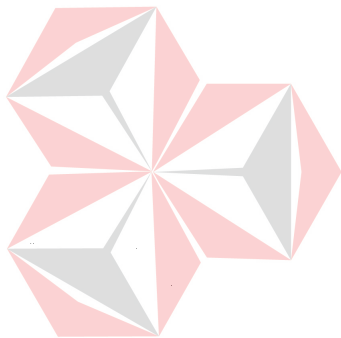
$$Kolom3 = Kolom1 - BarisDiff \quad \dots\dots\dots (persamaan 2.3)$$

$$Baris3 = Baris1 + KolomDiff \quad \dots\dots\dots (persamaan 2.4)$$

$$Kolom4 = Kolom2 - BarisDiff \quad \dots\dots\dots (persamaan 2.5)$$

$$Baris4 = Baris2 + KolomDiff \quad \dots\dots\dots (persamaan 2.6)$$

Rumus di atas dipakai oleh metode *Forward-Chaining* dan Teknik *Depth-First Search* untuk mencari kemungkinan kemenangan dan blok yang efektif untuk menghambat laju dari permainan lawan.



UNIVERSITAS
Dinamika

BAB III

PERMASALAHAN

Untuk mengetahui suatu masalah pada permainan yang menggunakan kemampuan berpikir, ketelitian dan kecerdasan seseorang dapat dilihat dari ide yang sederhana tetapi memiliki teknik dan metode pembuatan yang cukup tinggi. Seperti pada permainan Go-moku yaitu permainan sederhana yang menggunakan sebuah papan permainan dan alat tulis dimana pemain akan berusaha saling menjatuhkan / menghalangi untuk membentuk sebuah garis lurus dan siapa yang berhasil pertama kali membuatnya maka dinyatakan sebagai pemenangnya.

Tujuan tugas akhir ini adalah merancang dan membuat sebuah program permainan Quad yang merupakan kembangan dari permainan Go-moku yang selama ini permainan ini masih dimainkan secara manual.

Komponen utama dari permainan ini adalah sebuah papan permainan, 40 buah yang terdiri dari 20 merah dan 20 biru, 12 buah putih yang dibagi menjadi 6 buah putih untuk merah dan 6 buah putih yang lain untuk biru. Pada papan permainan terdapat lingkaran yang banyaknya 11 x 11 dan lingkaran pada setiap sudut nya dihilangkan.

Dalam program permainan komputer tampilan merupakan salah satu hal yang perlu diperhatikan, untuk dapat menarik perhatian seseorang memainkan permainan tersebut. Selain itu program permainan yang baik adalah program permainan yang mudah dimengerti oleh pemakai komputer (bersifat *user friendly*). Seperti bentuk papan permainan, informasi buah yang tersedia, jumlah kemengangan, seri, berapa

kali memainkannya dan efek suara serta informasi cara memainkannya perlu dicantumkan pada program tersebut.

Dalam program permainan Quad terdapat 2 jenis pemain yaitu manusia dan komputer. Pemain (manusia) pada umumnya mempunyai strategi sendiri-sendiri untuk menentukan posisi mana pada papan permainan yang akan ditempati oleh buah yang dipegangnya. Sedangkan untuk pemain lain (komputer) diperlukan suatu metode yang baik, efektif dan dapat membaca permainan lawan dengan berdasarkan pada metode Forward-Chaining dan teknik Depth-First.



UNIVERSITAS
Dinamika

BAB IV

IMPLEMENTASI DAN PEMBAHASAN

Dalam bab 4 ini akan dijelaskan program permainan Quad sesuai dengan metode, teknik yang dipakai dalam pembuatan program permainan komputer Quad dan aturan main dari permainan Quad tersebut serta perangkat pendukung yang digunakan.

4.1. Perangkat Pendukung.

Pemrograman berbasis windows pada umumnya memakai perangkat keras sebagai berikut :

- a. Komputer 486 DX2-66 Mhz atau lebih.
- b. Memori 4 Mb atau lebih.
- c. Monitor VGA / Super VGA.
- d. VGA Card 256 warna atau lebih.
- e. Dan perangkat lainnya.

4.2. Rancangan Perangkat Lunak.

Rancangan perangkat lunak yang dimaksudkan adalah desain sistem dari program berdasarkan teori dan aturan permainan Quad sendiri serta penjelasan procedure dan function dari program permainan Quad.

Algoritma dan Diagram Alir serta penjelasan program dari permainan komputer yang berdasarkan teori dan aturan permainan Quad adalah sebagai berikut :

4.2.1. Algoritma.

Algoritma yang dipakai oleh komputer (manusia lawan komputer) untuk menjalankan buah biru dengan memakai teknik pelacakan *Depth-First* dan metode *Forward-Chaining* seperti dibawah ini :

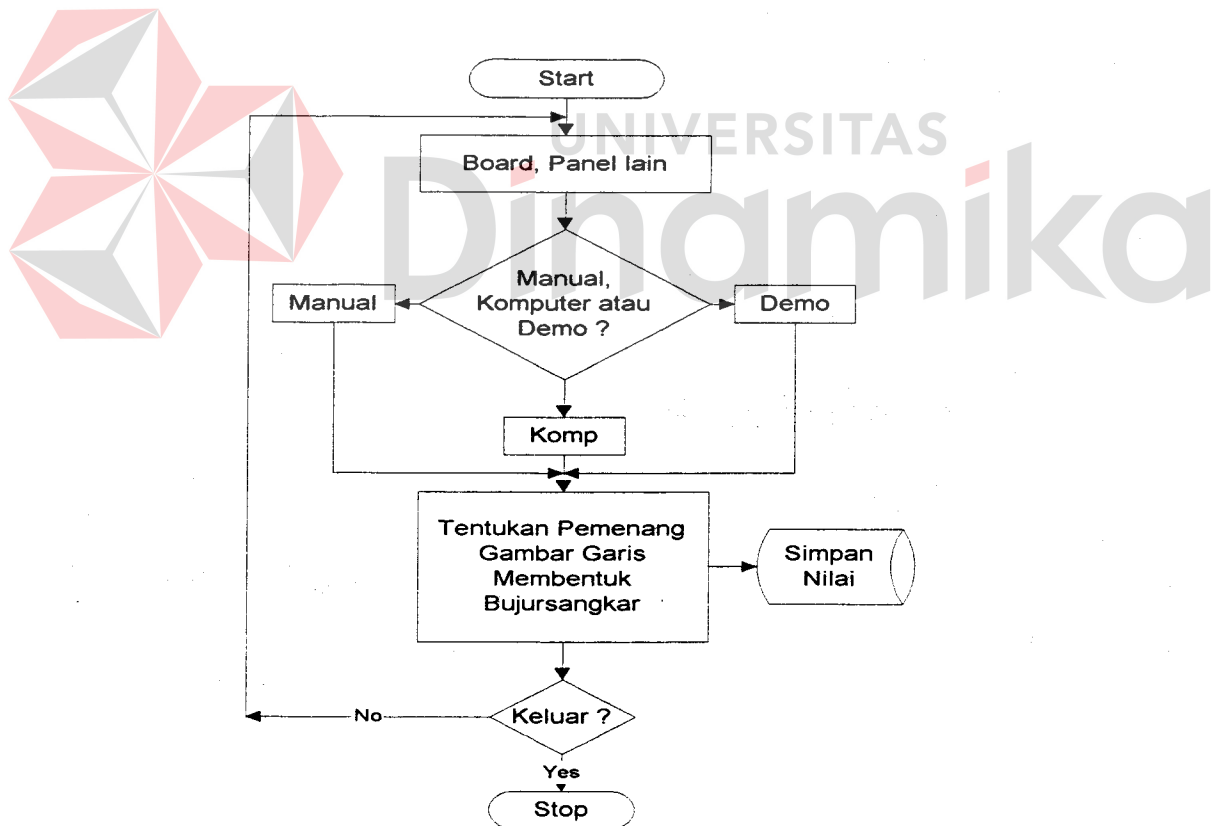
- a. Apakah lawan (buah merah) sudah jalan, jika belum tunggu sampai buah merah jalan.
- b. Letakkan buah biru pada sebarang tempat (secara acak).
- c. Scanning papan permainan dengan menggunakan teknik pelacakan *Depth-First* (yaitu baca dari awal board dengan pembacaan per baris) untuk mengetahui posisi lawan.
- d. Bila buah merah sudah mencapai tiga posisi untuk mencapai kemenangan maka lakukan blok agar langkah lawan terhenti.
- e. Bila tidak, cari kemungkinan langkah lawan berikutnya untuk mencapai kemenangan yang akan digunakan oleh lawan.
- f. Dari catatan dua kemungkinan posisi lawan yang kemungkinan menang letakkan buah biru secara acak.
- g. Jika sudah membentuk empat titik atau habis, tentukan pemenangnya, menang, seri atau kalah.
- h. Jika menang atau kalah beri garis dengan melalui ke empat titik tersebut yang pada akhirnya membentuk sebuah bujursangkar.
- i. Kembali ke no. 3.

Untuk algoritma manual (manusia lawan manusia) komputer hanya berperan untuk membatasi papan permainan dan cek apakah sudah mencapai empat titik, jika sudah bentuk garis dan tentukan siapa pemenangnya.

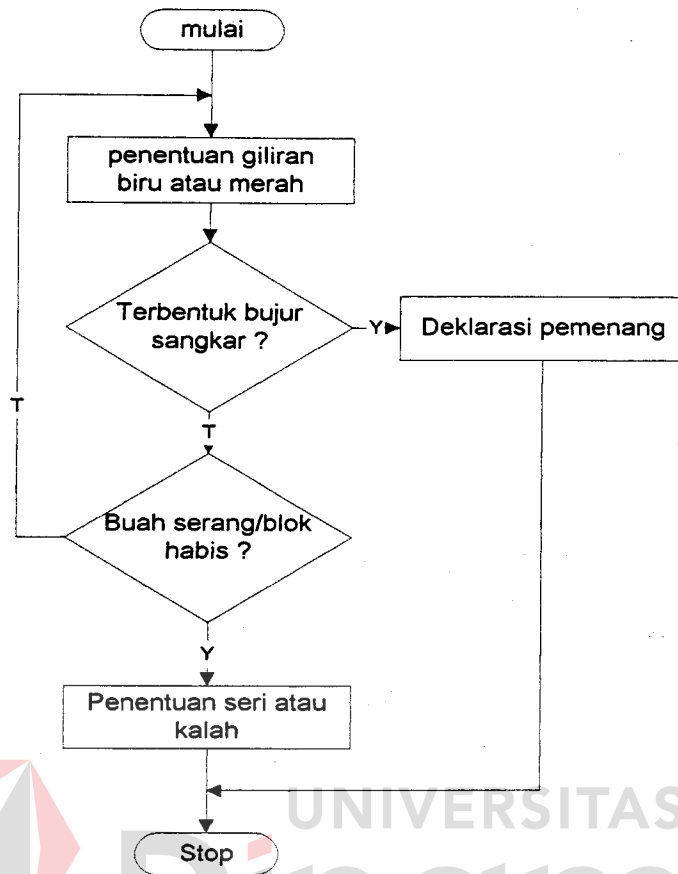
Sedangkan untuk Demo (komputer lawan komputer) algoritmanya sama dengan algoritma milik komputer juga memakai metode dan teknik yang sama. Dan buah merah dikendalikan oleh komputer juga.

4.2.2. Diagram alir program permainan Quad.

Secara garis besar program permainan Quad dapat di lihat pada gambar diagram alir dibawah ini :



Gambar 4.1. Diagram alir program utama permainan Quad

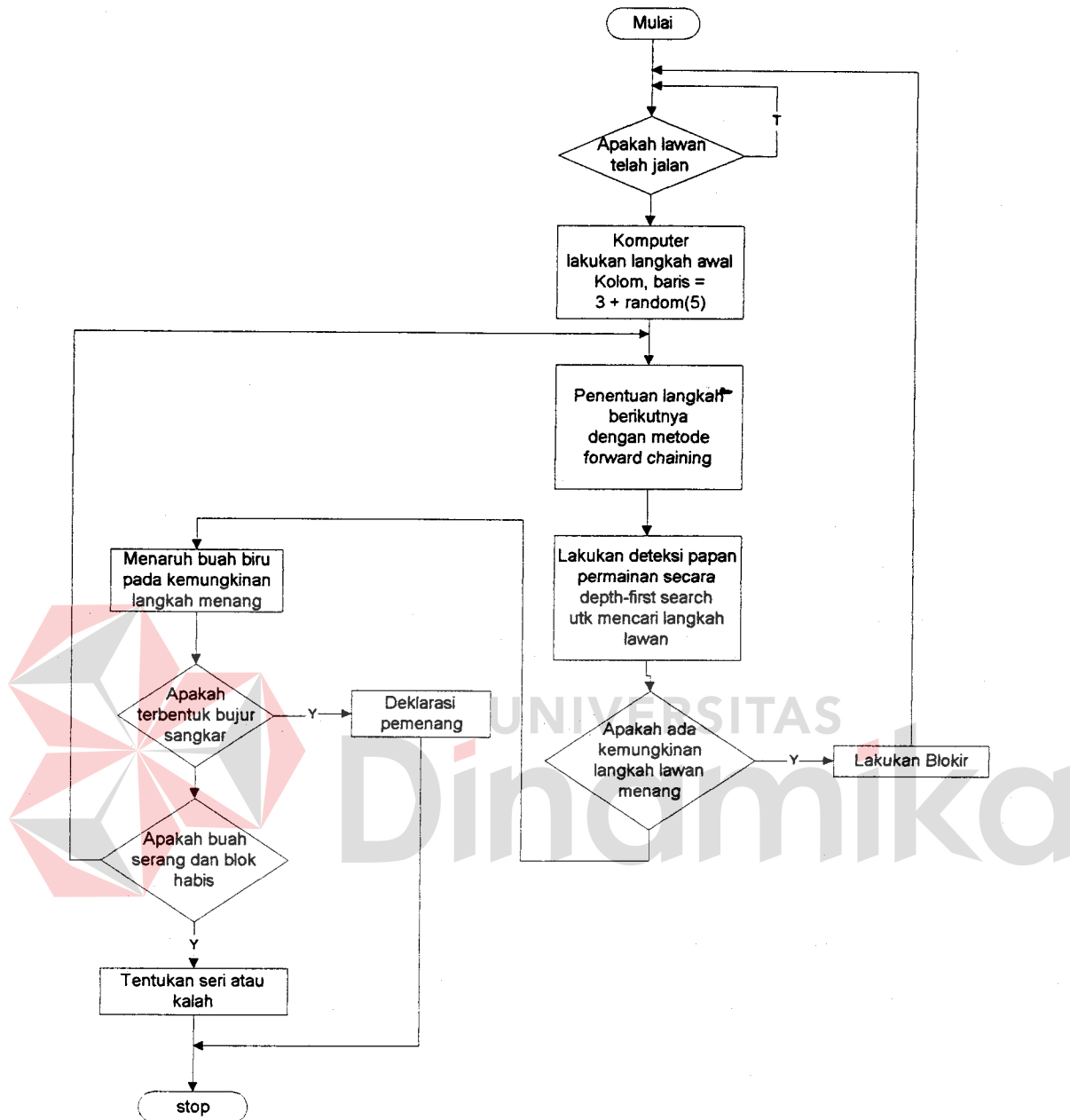


Gambar 4.2. Diagram alir mode manual



UNIVERSITAS

Dinamika



Gambar 4.3. Diagram alir mode komputer

Penentuan langkah berikutnya bagi komputer berdasarkan metode forward-chaining dan teknik pelacakan depth-first menggunakan rutin-rutin program sebagai berikut :

function TForm_Utama.Kemungkinan_Biru_Menang : boolean;

```

procedure TForm_Utama.Buat_Merah_Berjalan;
procedure TForm_Utama.Buat_Biru_Berjalan;
function TForm_Utama.Start_Potensial : boolean;
function TForm_Utama.Blokir_Potensial : boolean;
function TForm_Utama.Biru_Serang : boolean;
function TForm_Utama.Blokir_dan_Serang : boolean;
function TForm_Utama.Blokir_Dg_Blokir : boolean;
function TForm_Utama.Blokir_Ofensif : boolean;
procedure TForm_Utama.Move;
procedure TForm_Utama.Cek_Pemenang;

```

Berikut ini merupakan salah satu rutin yang memakai metode forward-chaining dan teknik pelacakan depth-first.



```

procedure TForm_Utama.Buat_Biru_Berjalan;
var
  Kolom1, Baris1, Kolom2, Baris2 : integer;
  Kolom3, Baris3, Kolom4, Baris4 : integer;
  KolomDiff, BarisDiff : integer;
begin
  BiruMovesCnt:= 0;
  for Kolom1:= 0 to 10 do
    for Baris1:= 0 to 10 do
      if Squares[Kolom1, Baris1] = Biru then
        for Kolom2:= 0 to 10 do
          for Baris2:= 0 to 10 do
            if (Squares[Kolom2, Baris2] = Biru) and
              ((Kolom1 <> Kolom2) or (Baris1 <> Baris2)) then
              begin
                KolomDiff:= Kolom2 - Kolom1;
                BarisDiff:= Baris2 - Baris1;
                Kolom3:= Kolom1 - BarisDiff;
                Baris3:= Baris1 + KolomDiff;
                Kolom4:= Kolom2 - BarisDiff;

```

```

    Baris4:= Baris2 + KolomDiff;
    if (Batas_Square(Kolom3, Baris3)) and
       (Batas_Square(Kolom4, Baris4)) and
       (Squares[Kolom3, Baris3] = Kosong) and
       (Squares[Kolom4, Baris4] = Kosong) then
    begin
        BiruMoves[BiruMovesCnt].Kolom:= Kolom3;
        BiruMoves[BiruMovesCnt].Baris:= Baris3;
        inc(BiruMovesCnt);
        BiruMoves[BiruMovesCnt].Kolom:= Kolom4;
        BiruMoves[BiruMovesCnt].Baris:= Baris4;
        inc(BiruMovesCnt);
    end;
end;
end;

```

Berdasarkan rutin diatas, teknik pelacakan depth-first yang berfungsi untuk melacak papan permainan dapat ditunjukkan pada potongan program berikut ini:

```

for Kolom1:= 0 to 10 do
for Baris1:= 0 to 10 do
if Squares[Kolom1, Baris1] = Biru then
for Kolom2:= 0 to 10 do
for Baris2:= 0 to 10 do
if (Squares[Kolom2, Baris2] = Biru) and
((Kolom1 <> Kolom2) or (Baris1 <> Baris2)) then

```

Sedangkan metode forward-chaining yang berfungsi untuk menentukan kemungkinan langkah menang berdasarkan langkah sebelumnya dapat dilihat pada potongan program sebagai berikut :

```

begin
    KolomDiff:= Kolom2 - Kolom1;
    BarisDiff:= Baris2 - Baris1;

```



```

Kolom3:= Kolom1 - BarisDiff;
Baris3:= Baris1 + KolomDiff;
Kolom4:= Kolom2 - BarisDiff;
Baris4:= Baris2 + KolomDiff;
if (Batas_Square(Kolom3, Baris3)) and
   (Batas_Square(Kolom4, Baris4)) and
   (Squares[Kolom3, Baris3] = Kosong) and
   (Squares[Kolom4, Baris4] = Kosong) then
begin
  BiruMoves[BiruMovesCnt].Kolom:= Kolom3;
  BiruMoves[BiruMovesCnt].Baris:= Baris3;
  inc(BiruMovesCnt);
  BiruMoves[BiruMovesCnt].Kolom:= Kolom4;
  BiruMoves[BiruMovesCnt].Baris:= Baris4;
  inc(BiruMovesCnt);
end;
end;

```

4.2.3. Penjelasan program.

Dalam membuat program permainan Quad ini, bahasa pemrograman Borland Delphi sangat berperan dalam membantu penulisan program. Karena interface yang diperlukan tinggal diambil pada toolbar dan untuk pengendalian interfacenya terletak pada rutin-rutin.

A. Rutin pembuatan papan permainan.

Papan permainan dibagi menjadi dua bagian, bagian pertama adalah papan permainan itu sendiri dan bagian kedua adalah panel informasi. Panel informasi berisi waktu sekarang dengan memanggil rutin *procedure TForm_Utama.Timer3Timer(Sender: TObject);*, pemain yang akan jalan dengan memanggil rutin *procedure TForm_Utama.UpdatePlayer;*, buah permainan dengan memanggil rutin *procedure*

TForm_Utama.BoardMouseDown(Sender: TObject; Button: TMouseButton; Shift: TShiftState; X, Y: Integer);, dan informasi Nilai kemenangan tiap pemain, seri, berapa kali bermain dengan memanggil rutin *procedure TForm_Utama.Update_Pemenang;*.

Untuk membuat gambar papan permainan, dideklarasikan terlebih dahulu type, image, besarnya lingkaran dan ukuran papan permainan itu sendiri. Rutin pemanggilannya sebagai berikut :

```
Private
const
    SquareSize      = 34;
    QuadSize        = 24;
    QuadStart       = (SquareSize - QuadSize) div 2 + 1;
    Center           = SquareSize div 2 + 1;
    StatusSpacing   = 8;
```

Berdasarkan konstanta di atas maka pembuatan papan permainan dan buah menjadi lebih mudah dan ukurannya dapat dirubah. Adapun rutin pemanggilnya adalah sebagai berikut :

```
Procedure TForm_Utama.DrawBoard;
```

Algoritma prosedur di atas adalah sebagai berikut :

1. Kolom 0 - 10
2. Baris 0 -10
3. Jika sesuai dengan Batas_Square, Draw_Quad(Kolom, Baris)

Sedangkan untuk membuat buah permainan digunakan rutin sebagai berikut :

```
Procedur TForm_Utama.Draw_Quad(Kolom, Baris :integer);
```

Algoritma yang dipakai adalah :

$$X = Kolom * QuadSize + QuadStart$$

$$Y = Baris * QuadSize + QuadStart$$

Tentukan warna buah permainan dan lingkaran

Gambar Buah Quad : $Ellipse(x, y, x+QuadSize, y+QuadSize)$

B. Rutin batas papan permainan.

Rutin tersebut untuk membatasi letak dari lingkaran sesuai dengan aturan main Quad. Batasnya adalah mulai dari kolom 0 sampai kolom 10 dan baris 1 sampai dengan baris 9, atau untuk kolom 1 sampai dengan kolom 9, baris 0 sampai dengan baris 10.

Rutin pemanggilannya adalah *function Batas_Square(Kolom, Baris : integer) : boolean;*

C. Pengendalian strategi komputer (buah biru).

C.1. Langkah awal komputer.

Komputer melakukan langkah awal dengan mempertimbangkan bahwa suatu posisi baik adalah posisi yang mampu membentuk banyak kemungkinan langkah berikutnya dalam arti membentuk suatu bujursangkar sebagai tujuan /goal. Rutin yang dipanggil adalah *function TForm_Utama.Langkah_Awal_Komp : boolean;*

Algoritma rutin di atas adalah sebagai berikut :

Untuk menentukan posisi baris dan kolom secara acak dengan batasan sebagai berikut, $3 + \text{random}(5)$. Batasan tersebut dimaksudkan untuk menempatkan posisi tengah, dimana posisi tersebut mempunyai banyak kemungkinan membentuk bujursangkar.

C.2. Penentuan langkah komputer selanjutnya.

Untuk menentukan langkah komputer selanjutnya berdasarkan langkah yang dilakukan sebelumnya dan langkah lawan. Rutin yang dipanggil adalah *function TForm_Utama.Kemungkinan_Biru_Menang : boolean;*.

Algoritma rutin tersebut adalah sebagai berikut :

Lakukan pembacaan papan permainan secara Depth-First Search dan catat langkah awal sebelumnya, lakukan perhitungan untuk menentukan langkah kemenangan berdasarkan langkah lawan.

C.3. Deteksi kemungkinan langkah lawan menang.

Untuk mencari mengantisipasi terjadinya kemungkinan langkah lawan menang, maka perlu dilakukan pengecekan langkah lawan sebelumnya serta mencari kemungkinan menang yang akan dibentuk oleh lawan. Rutin yang dipanggil adalah *procedure TForm_Utama.Buat_Merah_Berjalan;*

Algoritma dari rutin di atas adalah sebagai berikut :

Lakukan pembacaan papan permainan secara Depth-First Search dan catat langkah awal sebelumnya, lakukan perhitungan untuk menentukan langkah kemenangan lawan.

C.4. Deteksi kemungkinan langkah komputer menang.

Untuk mencari mencari kemungkinan langkah menang, maka perlu dilakukan pengecekan langkah sebelumnya serta mencatat kemungkinan langkah menang komputer. Rutin yang dipanggil adalah *procedure TForm_Utama.Buat_Biru_Berjalan;*

Algoritma dari rutin di atas adalah sebagai berikut :

Lakukan pembacaan papan permainan secara Depth-First Search dan catat langkah sebelumnya, lakukan perhitungan untuk menentukan langkah kemenangan.

C.5. Strategi blok.

Dalam strategi blok dibagi menjadi beberapa function yang kegunaannya berbeda.

C.5.1. Blok dan serang.

Strategi blok ini digunakan untuk mengantisipasi kemenangan lawan sekaligus membentuk langkah berikutnya dengan cara melakukan blok dengan buah penyerang (biru). Rutin yang dipanggil adalah *function TForm_Utama.Blokir_Ofensif : boolean; .*

Algoritma dari rutin di atas adalah :

Jika kemungkinan langkah menang lawan sama dengan langkah komputer berikutnya maka komputer akan meletakkan buah penyerangnya (biru) pada posisi tersebut.

C.5.2. Blok dengan buah putih.

Untuk menutupi kemenangan lawan dilakukan blokir dengan buah putih. Rutin yang dipanggil adalah *function TForm_Utama.Blokir_Dg_Blokir : boolean;*

Algoritma dari *function* di atas adalah jika kemungkinan langkah menang lawan sama dengan langkah komputer berikutnya maka komputer akan meletakkan buah penyerangnya (biru) pada posisi tersebut.

D. Rutin pengendalian mouse.

Untuk memakai program permainan ini mouse sangat diperlukan sehingga pengendaliannya harus dimasukkan dalam program permainan Quad, adapun

pemanggilan rutinnnya adalah *procedure TForm_Utama.BoardMouseDown (Sender: TObject; Button: TMouseButton; Shift: TShiftState; X, Y: Integer);*. Dengan logikanya adalah jika tekan tombol mouse kanan buah putih yang dijalankan, sebaliknya tombol kiri mouse yang ditekan buah permainan yang dijalankan dengan kondisi memenuhi batas dari permainan.

E. Menentukan pemenang dan garis Quad.

Rutin untuk menentukan pemenang permainan dengan memanggil

Procedure TForm_Utama.Cek_Pemenang;

Dengan algoritma sebagai berikut :

Jika buah merah dan biru habis

Mulai perbandingan buah blok milik merah dengan biru, yang paling banyak adalah pemenangnya, selain itu seri.

Untuk pemenang maka komputer akan membuat garis bujursangkar yang menghubungkan keempat buah titik tersebut. Jika pemenangnya merah maka garis berwarna merah, jika pemenangnya biru maka garisnya berwarna biru.

F. Menyimpan nilai kemenangan.

Untuk menyimpan nilai kemenangan dari semua pemain, berapa kali seri, mode permainan dengan memanggil :

Procedure TForm_Utama.FormDestroy(Sender : TObject);

Dengan memakai file yang berinisial *.ini dan nama filenya diambil dari *.exe permainan, tulis Mode Komputer untuk Option, Nilai dari Merah, Biru dan Seri berapa kali.

G. Mengendalikan lagu / suara background.

Rutin untuk mengendalikan suara / lagu dengan membuat form dan dijalankan secara bersama-sama dengan form utama. Secara umum untuk memanggil suara / lagu dengan menentukan tipe dari lagu / suara diantaranya adalah *.mid atau *.wav (dtSequencer atau dtWaveAudio), nama file, buka file dan jalankan file tersebut.

Misalkan panggil suara tepuk tangan dengan tipe *.wav.

```
With MediaPlayer1 do
Begin
    DeviceType:=dtWaveAudio;
    FileName:='win5.wav';
    Open;
    Play;
End;
```

4.3. Pengujian.

Berdasarkan penjelasan diagram alir dan algoritma program di atas, untuk lebih detail tentang perhitungan rumus yang dipakai metode *Forward-Chaining* dan teknik *Depth-First* sebagai strategi komputer dalam menentukan posisi buah ke-2 dan ke-3 seperti contoh dibawah ini :

Posisi buah merah 1 terletak pada kolom 03, baris 05

 2 terletak pada kolom 04, baris 02

Untuk mengetahui letak posisi buah biru ke-2 dan ke-3 adalah sebagai berikut :

```
BarisDiff        = Baris 2 – Baris 1
                  = 02 – 05
                  = - 03

KolomDiff        = Kolom 2 – Kolom 1
                  = 04 – 03
                  = 01
```

$$\begin{aligned}\text{Kolom 3} &= \text{Kolom 1} - \text{BarisDiff} \\ &= 03 - (-03) \\ &= 06\end{aligned}$$

$$\begin{aligned}\text{Baris 3} &= \text{Baris 1} + \text{KolomDiff} \\ &= 05 + (01) \\ &= 06\end{aligned}$$

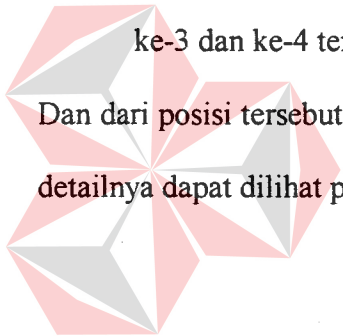
$$\begin{aligned}\text{Kolom 4} &= \text{Kolom 2} - \text{BarisDiff} \\ &= 04 - (-03) \\ &= 07\end{aligned}$$

$$\begin{aligned}\text{Baris 4} &= \text{Baris 2} + \text{KolomDiff} \\ &= 02 + 01 \\ &= 03\end{aligned}$$

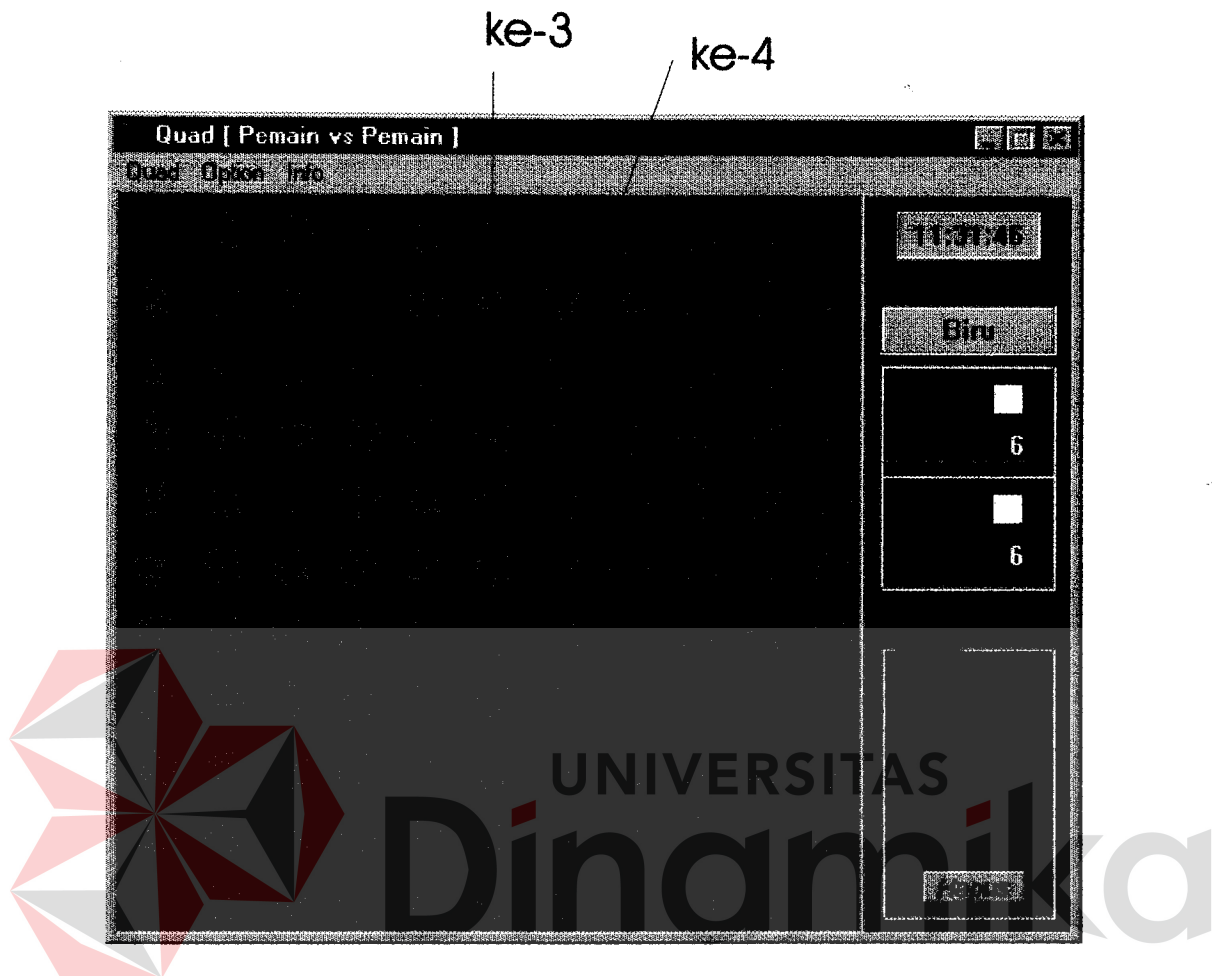
Berdasarkan perhitungan di atas, posisi kemungkinan kemenangan milik buah merah

ke-3 dan ke-4 terdapat pada kolom 06, baris 06 dan kolom 07, baris 03.

Dan dari posisi tersebut komputer dapat menentukan langkah berikutnya. Untuk lebih detailnya dapat dilihat pada gambar dibawah ini :



UNIVERSITAS
Dinamika



Gambar 4.4. Kemungkinan merah menang

Setelah komputer meletakkan buah biru maka akan dianalisa apakah buah merah berikutnya sudah jalan, jika sudah maka perhitungan selanjutnya diteruskan berdasarkan pada buah biru sesuai dengan metode *Forward-Chaining*, analisa terhadap pemain merah dengan teknik *Depth-First* seperti penjelasan di atas.

Untuk mengetahui lebih jauh apakah program yang telah dibuat sesuai dengan metode yang dipilih yaitu forward-chaining dan depth-first, maka perlu dilakukan

beberapa pengujian permainan . Adapun hasil pengujiannya dapat dilihat pada tabel berikut ini :

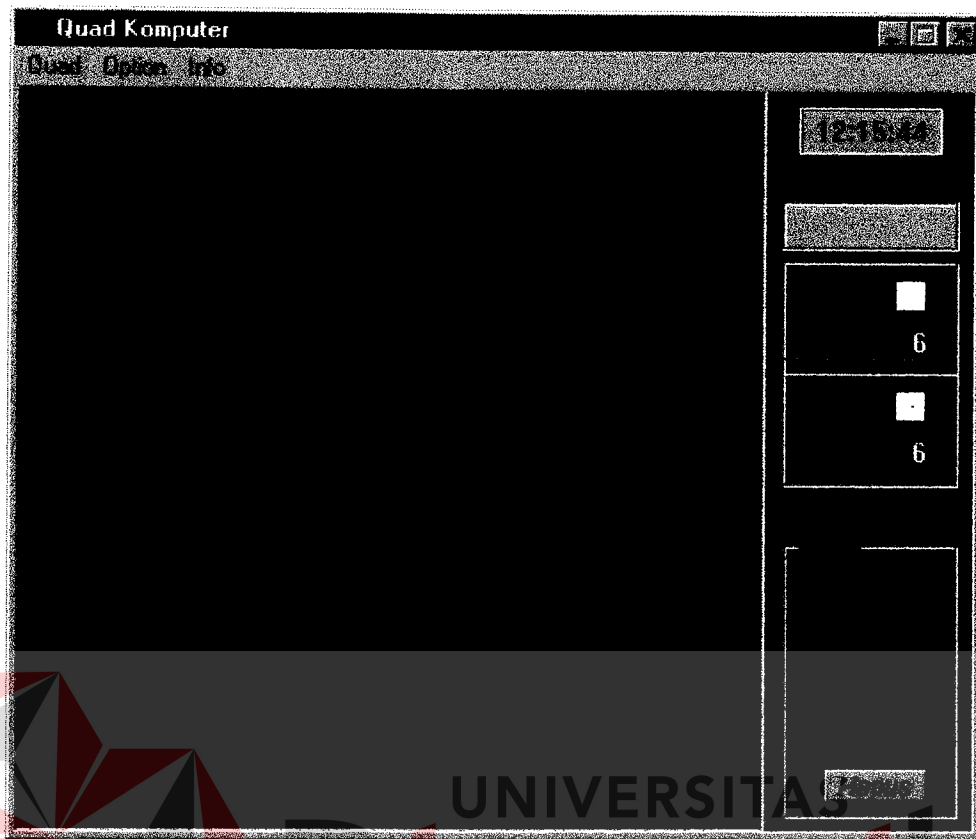
Tabel 4.1. Percobaan 1 pengujian strategi komputer

Step	Analisa											
	Komputer				Perhitungan Manual				Lawan			
	Serang		Blok		Serang		Blok		Serang		Blok	
	K	B	K	B	K	B	K	B	K	B	K	B
1	3	5	-	-	-	-	-	-	2	2	-	-
2	0	4	-	-	0	4	-	-	4	4	-	-
3	4	2	-	-	4	2	-	-	3	2	-	-
4	1	1	-	-	1	1	-	-	4	3	-	-

Keterangan :

Pada Analisa untuk semua tabel terdapat Komputer, Perhitungan Manual dan Lawan, masing-masing akan dijelaskan dibawah ini :

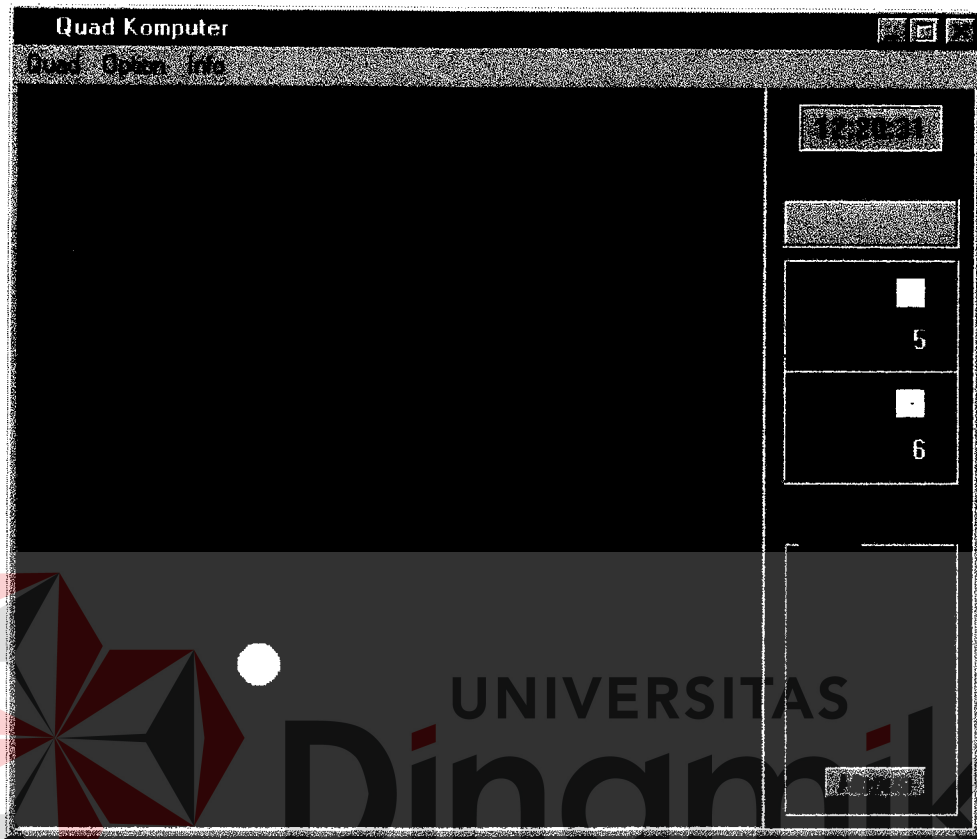
1. Komputer dalam menjalankan buah permainan memegang buah biru.
2. Lawan adalah manusia yang memegang buah merah.
3. Perhitungan Manual adalah jalannya komputer yang dihitung untuk menentukan posisinya berdasarkan rumus dan dihitung secara manual.



Gambar 4.5. Percobaan 1

Tabel 4.2. Percobaan 2 pengujian strategi komputer

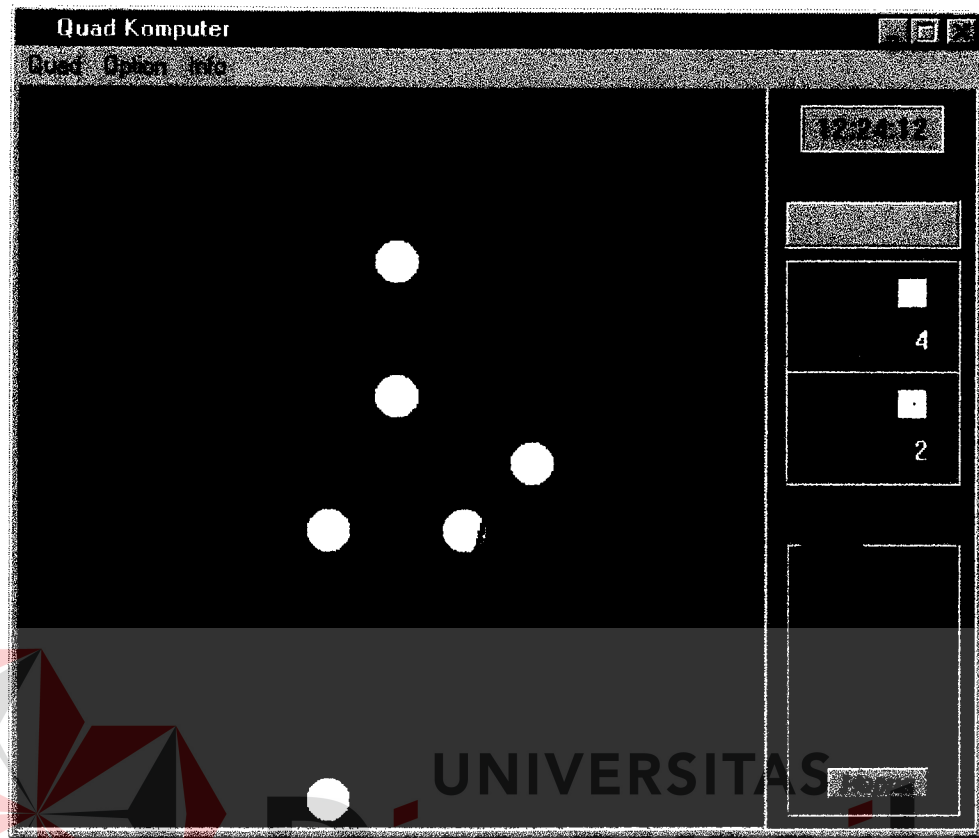
Step	Analisa											
	Komputer				Perhitungan Manual				Lawan			
	Serang		Blok		Serang		Blok		Serang		Blok	
	K	B	K	B	K	B	K	B	K	B	K	B
1	5	5	-	-	-	-	-	-	7	9	-	-
2	9	7	-	-	9	7	-	-	5	7	-	-
3	6	5	-	-	6	5	-	-	7	8	-	-
4	7	0	-	-	7	0	-	-	6	7	-	-
5	-	-	-	-	-	-	-	-	-	-	3	8
6	4	8	-	-	4	8	-	-	6	8	-	-



Gambar 4.6. Percobaan 2

Tabel 4.3. Percobaan 3 pengujian strategi komputer

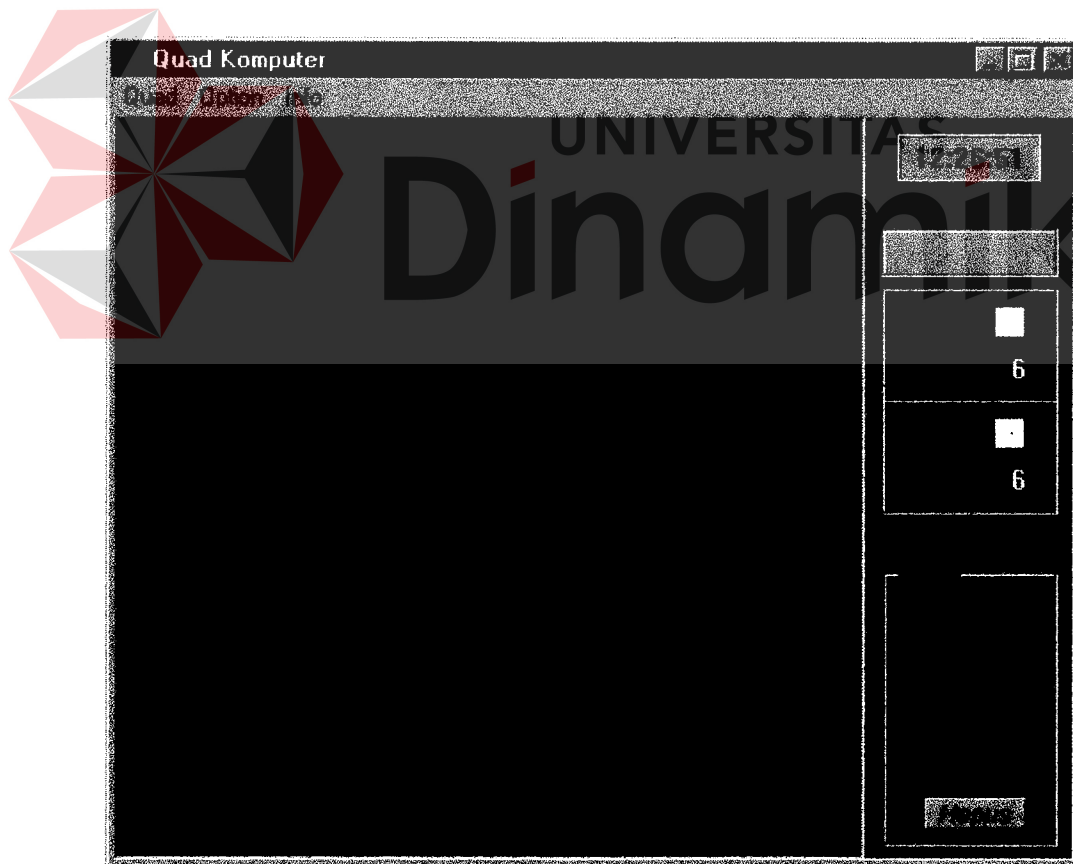
Step	Analisa											
	Komputer				Perhitungan Manual				Lawan			
	Serang		Blok		Serang		Blok		Serang		Blok	
	K	B	K	B	K	B	K	B	K	B	K	B
1	3	5	-	-	-	-	-	-	5	5	-	-
2	6	7	-	-	6	7	-	-	7	4	-	-
3	1	8	-	-	1	8	-	-	6	5	-	-
4	-	-	-	-	-	-	-	-	-	-	4	10
5	-	-	-	-	-	-	-	-	6	4	-	-
6	-	-	7	5	-	-	7	5	-	-	-	-
7	-	-	5	4	-	-	5	4	-	-	-	-
8	8	4	-	-	8	4	-	-	-	-	5	2
9	-	-	-	-	-	-	-	-	5	6	-	-
10	-	-	6	6	-	-	6	6	-	-	-	-
11	7	3	-	-	7	3	-	-	4	5	-	-
12	-	-	4	6	-	-	4	6	-	-	-	-
13	2	6	-	-	2	6	-	-	4	7	-	-
14	3	2	-	-	3	2	-	-	-	-	-	-



Gambar 4.7. Percobaan 3

Tabel 4.4. Percobaan 4 pengujian strategi komputer

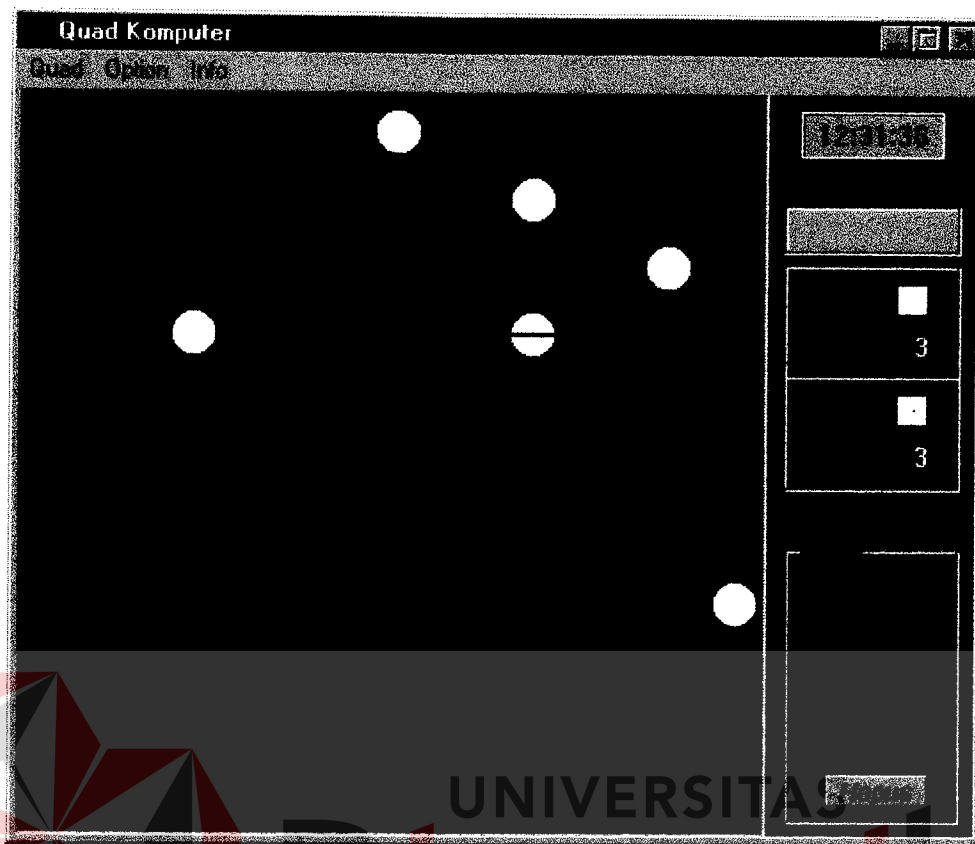
Step	Analisa											
	Komputer				Perhitungan Manual				Lawan			
	Serang		Blok		Serang		Blok		Serang		Blok	
	K	B	K	B	K	B	K	B	K	B	K	B
1	4	4	-	-	-	-	-	-	9	10	-	-
2	7	6	-	-	7	6	-	-	6	9	-	-
3	2	7	-	-	2	7	-	-	8	9	-	-
4	1	2	-	-	1	2	-	-	5	9	-	-
5	6	1	-	-	6	1	-	-	6	10		



Gambar 4.8. Percobaan 4

Tabel 4.5. Percobaan 5 pengujian strategi komputer

Step	Analisa											
	Komputer				Perhitungan Manual				Lawan			
	Serang		Blok		Serang		Blok		Serang		Blok	
	K	B	K	B	K	B	K	B	K	B	K	B
1	6	3	-	-	-	-	-	-	9	1	-	-
2	6	7	-	-	6	7	-	-	7	4	-	-
3	2	7	-	-	2	7	-	-	7	2	-	-
4	-	-	-	-	-	-	-	-	-	-	2	3
5	10	3	-	-	10	3	-	-	8	1	-	-
6	-	-	-	-	-	-	-	-	-	-	10	7
7	-	-	9	2	-	-	9	2	8	2	-	-
8	-	-	7	1	-	-	7	1	-	-	-	-
9	9	3	-	-	9	3	-	-	8	4	-	-
10	2	4	-	-	2	4	-	-	-	-	5	0
11	-	-	7	3	-	-	7	3	8	3	-	-
12	6	6	-	-	6	6	-	-	6	0	-	-
13	9	6	-	-	9	6	-	-	-	-	-	-



Gambar 4.9. Percobaan 5

4.4. Analisa Hasil Pengujian.

Berdasarkan pengujian yang telah dilakukan sebanyak lima kali percobaan, terlihat bahwa antara hasil perhitungan yang menggunakan metode forward-chaining dan teknik pelacakan depth-first dalam program permainan Quad sesuai dengan perhitungan secara manual. Dan strategi yang dipakai komputer terbukti efektif dan tangguh dalam menentukan posisinya untuk mencapai kemenangan.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan pengujian dan analisa hasil pengujian yang telah dipaparkan pada bab sebelumnya dapat disimpulkan sebagai berikut :

- a. Telah dapat dirancang dan dibuat program permainan Quad menggunakan algoritma Forward-Chaining dan metode pelacakan Depth-First Search dengan benar dan sesuai dengan aturan permainan Quad yang sebenarnya.
- b. Permainan tersebut dapat dipakai / digunakan untuk satu atau dua orang pemain.
- c. Program permainan Quad ini dapat digunakan pada komputer PC dengan spesifikasi perangkat keras yang sesuai dengan sistem operasi windows 95.

5.2. Saran

Adapun saran yang diperlukan demi penyempurnaan program permainan Quad ini adalah sebagai berikut :

- a. Perlu ditambahkan tingkat kesulitan pada permainan Quad supaya lebih menyenangkan dan tidak membuat bosan pemakainya.
- b. Programnya agar dapat dibuat dan diaplikasikan dalam suatu jaringan komputer atau kalau bisa dapat dimainkan dalam jaringan internet.
- c. Metode kecerdasan buatan dan teknik pelacakan di sempurnakan lagi, bila perlu digabung agar tingkat kesulitan lebih tinggi.

DAFTAR PUSTAKA

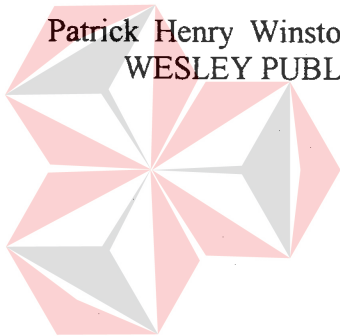
Bill Todd and Vince Kellen with Ray Novak dan Brad Saenz, 1995, Delphi A Developer's Guide, M&T Books.

Charles Calvert, 1996, Delphi 2 Unleashed, Second Edition, SAMS Publishing.

Herbert Schildt, 1987, ARTIFICIAL INTELLIGENCE USING C, Osborne McGraw-Hill.

Inside Visual Basic for Windows, Vol. 7 No. 2, February 1997, The COBB Group.

Patrick Henry Winston, 1992, Artificial Intelligence, Third Edition, ADDISON-WESLEY PUBLISHING COMPANY.



UNIVERSITAS
Dinamika