**LAPORAN PRAKTIKUM KECERDASAN BUATAN**

**“KECERDASAN BUATAN GAME SERIGALA DOMBA”**

****

**Disusun Oleh:**

**KELOMPOK B/5/2015**

Novreza Herdian L 1515015079

Muhammad Asyharul Ma’arif 1515015078

Dina Rusmawati 1515015071

Willyardo Tampubolon 1515015069

**Asisten Praktikum :**

M. Hilmy Ady Saputra Niken Novirasari

1415015058 1415015064

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI DAN KOMUNIKASI**

**FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI**

**UNIVERSITAS MULAWARMAN**

**2017**

**KATA PENGANTAR**

Puji Syukur senantiasa kita panjatkan kehadirat Tuhan Yang Maha Esa karena berkat Rahmat dan Izin-Nya jualah maka Laporan Akhir Praktikum ini dapat diselesaikan tepat waktu. Adapun tujuan dalam penulisan Laporan ini adalah untuk memenuhi tugas akhir Praktikum Kecerdasan Buatan pada semester empat.

Laporan ini terdiri dari beberapa bab yang bertujuan untuk memberikan pemaparan tentang projek akhir yang berjudulKecerdasan Buatan Game Serigala Domba. Dan dalam situasi Laporan ini pada awalnya penulis mengalami beberapa kesulitan dan terutama karena kurangnya pengetahuan, namun berkat bimbingan dan dukungan dari berbagai pihak sehingga Laporan ini dapat terselesaikan walaupun masih ada kekurangannya.

Dalam penulisan Laporan Akhir ini, penulis dengan kerendahan dan penuh pengharapan mengucapkan terima kasih kepada pihak-pihak yang membantu dalam menyelesaikan Laporan ini, khususnya kepada :

1. Tuhan Yang Maha Kuasa atas limpahan Rahmat-Nya.
2. Ibu Joan Angelina W, M.Kom dan Ibu Masna Wati, M.T selaku dosen mata kuliah Kecerdasan Buatan..
3. Asisten Laboratorium Praktikum Kecerdasan Buatan yang telah membimbing dalam pembuatan program dan penyelesaian Laporan Akhir.

Penulis telah berusaha untuk membuat Laporan Akhir ini dengan sebaik-baiknya, meskipun demikian penulis yakin jika Laporan ini masih jauh dari sempurna. Untuk itu penulis sangat terbuka untuk menerima kritik dan saran yang bersifat membangun. Terimakasih.

Samarinda, 3 Mei 2017

Penyusun

**DAFTAR ISI**

**Halaman Judul**

**Kata Pengantar**  i

**Daftar Isi** ii

**Daftar Tabel** iv

**Daftar Gambar** v

**BAB I Pendahuluan**  1

* 1. Latar Belakang 1
  2. Rumusan Masalah 1
  3. Batasan Masalah 1
  4. Tujuan dan Manfaat 2

**BAB II Landasan Teori** 3

1. Fungsi-Fungsi Visual Prolog Yang Digunakan 3
2. Unifikasi dan Lacakbalik 3
3. Data *Object* Sederhana dan Jamak 3
4. Perulangan dan Rekursi 4
5. *List* 4
6. *Section Facts* 5

2.2 Tentang Game Serigala Domba 5

**BAB III Metodologi** 8

1. Alur Pembuatan Sistem 8

**BAB IV Hasil dan Pembahasan**   **14**

1. Tabel Kebenaran 14

4.1.1 Tabel Kebenaran 14

4.1.2 Pohon Keputusan 15

1. Analisis Aplikasi 15

**BAB** **V Penutup** 23

1. Kesimpulan 23
2. Saran 23

**Daftar Pustaka**   **24**

**Lampiran**   **25**

**DAFTAR TABEL**

4.1.1.1 Tabel Kebenaran (1) 14

4.1.1.2 Tabel Kebenaran (2) 14

4.1.1.3 Tabel Kebenaran (3) 14

**DAFTAR GAMBAR**

Gambar 4.1.2.1 Pohon Keputusan 15

Gambar 3.1 Alur permainan 8  
Gambar 3.2 Urutan Permainan 9  
Gambar 3.3 Urutan Permainan 10  
Gambar 3.4 Kredit 11  
Gambar 3.5 Peraturan 12  
Gambar 3.6 Exit 13

Gambar 3.6 Exit 13

Gambar 4.2.1 Tampilan Awal 15

Gambar 4.2.2 Tampilan Credit 16

Gambar 4.2.3 Tampilan Peraturan 17

Gambar 4.2.4 Tampilan Mulai Game 17

Gambar 4.2.5 Tampilan Game Berjalan 18

Gambar 4.2.6 Tampilan Game Win 19

Gambar 4.2.7 Tampilan Game Over 20

Gambar 4.2.8 Tampilan Bermain Lagi 21

Gambar 4.2.9 Tampilan Kembali Ke Menu Utama 21

Gambar 4.2.10 Tampilan Exit 22

Gambar 4.2.11 Tampilan Perulangan Pada Menu Utama 22

**BAB I**

**PENDAHULUAN**

**1.1 Latar Belakang**

Permainan (game) yang semula dimainkan secara nyata, sekarang telah bisa dimainkan secara maya melalui program komputer. Kemajuan teknologi informasi dalam bidang kecerdasan buatan menjadi pemicu hadirnya game dalam sebuah komputer. Game menjadi sarana untuk melepaskan lelah dan juga melatih logika pemain game itu sendiri. Saat ini telah banyak jenis program game baik secara online, offline, hanya bisa dimainkan sendiri, maupun bisa dimainkan dengan lawan dalam sebuah jaringan komputer.

Salah satu jenis game yang sederhana namun cukup mengasah otak adalah permainan Serigala Domba. Dalam permainan logika ini terdapat serigala, kambing, sayuran dan petani. Diceritakan ada seorang petani yang hendak menyeberangi sungai membawa hasil belanjanya dari pasar, yaitu sekeranjang penuh sayuran, juga seekor serigala dan seekor domba. Pemain diminta untuk menyeberangkan petani, serigala, domba, dan sayuran menggunakan sebuah perahu yang hanya muat ditempati oleh dua penumpang. Yang dapat menggunakan perahu hanya sang petani. Permasalahnya adalah pada saat petani tidak ada, serigala akan memakan domba, dan domba akan memakan sayuran.

* 1. **Rumusan Masalah**

1. Bagaimana caranya petani menyeberangkan semua bawaannya?
2. Bagaimana pencarian solusi permainan serigala domba dengan menggunakan aplikasi visual prolog?
   1. **Batasan Masalah**

Agar pembahasan tidak menyimpang dari pokok rumusan masalah, penulis membatasi permasalahan pada :

1. Dalam pembuatan program ini menggunakan bahasa pemrograman prolog.
2. Program yang digunakan adalah Visual Prolog.
3. Program ini akan mencari solusi dari game Serigala Domba.
   1. **Tujuan dan Manfaat**

Tujuan dan manfaat dari penulisan laporan ini adalah untuk merancang dan membangun sebuah program berbasis game menggunakan bahasa Prolog.

Manfaat dari makalah ini adalah menambah ragam permainan yang telah ada sehingga dapat digunakan sebagai salah satu media alternatif untuk mengisi waktu senggang. Selain itu, permainan ini juga termasuk salah satu jenis permainan edukasi sehingga dapat digunakan untuk melatih kemampuan nalar dan logika seseorang.

**BAB II**

**LANDASAN TEORI**

1. **Fungsi-Fungsi Visual Prolog**
2. **Unifikasi dan Lacakbalik**

Pada waktu Visual Prolog mencoba untuk mencocokkan suatu panggilan (dari sebuah *subgoal*) ke klausa (pada *section clauses*), maka proses tersebut melibatkan suatu prosedur yang dikenal dengan unifikasi (*unification*), yang mana berusaha untuk mencocokkan antara struktur data yang ada di panggilan (*subgoal*) dengan klausa yang diberikan.

Unifikasi pada Prolog mengimplementasikan beberapa prosedur yang juga dilakukan oleh beberapa bahasa tradisional seperti melewatkan parameter, menyeleksi tipe data, membangun struktur, mengakses struktur dan pemberian nilai (*assignment*).

Pada intinya unifikasi adalah proses untuk mencocokkan dua predikat dan memberikan nilai pada variabel yang bebas untuk membuat kedua predikat tersebut identik. Mekanisme ini diperlukan agar Prolog dapat mengidentifikasi klausa-klausa mana yang dipanggil dan mengikat (*bind*) nilai klausa tersebut ke variabel.

1. **Data Object Sederhana dan Jamak**

Pada waktu menyelesaikan masalah, seringkali, seseorang harus menelusuri suatu jalur untuk mendapatkan konklusi yang logis. Jika konklusi ini tidak memberikan jawaban yang dicari, orang tersebut harus memilih jalur yang lain.

Metode balik-ke-atas-dan-coba-lagi (*backing-up-and-trying-again*) ini pada Visual Prolog disebut lacakbalik (*backtracking*).Visual Prolog menggunakan metode ini untuk menemukan suatu solusi dari permasalahan yang diberikan.Visual Prolog dalam memulai mencari solusi suatu permasalahan (atau *goal*) harus membuat keputusan di antara kemungkinan-kemungkinan yang ada.Ia menandai di setiap percabangan (dikenal dengan titik lacak balik) dan memilih *subgoal* pertama untuk telusuri. Jika *subgoal* tersebut gagal (ekuivalen dengan menemukan jalan buntu), Visual Prolog akan lacakbalik ke titik lacakbalik (*back-tracking point*) terakhir dan mencoba alternatif *subgoal* yang lain.

1. **Perulangan dan Rekursi**

Komputer memiliki bermacam kemampuan yang berguna salah satunya adalah kemampuan melakukan sesuatu berulang-ulang.Prolog dapat melakukan perulangan dalam dua hal yaitu berupa prosedur dan struktur data. Ide dari struktur data repetitif (rekursif) adalah bagaimana menciptakan struktur data yang ukuran (*size*) akhirnya belum diketahui ketika struktur tersebut pertama kali dibuat (*create*).

Prolog menyediakan dua jenis perulangan yaitu :

* 1. Lacakbalik

Ketika suatu prosedur melakukan lacakbalik, prosedur akan mencari alternatif jawaban dari sebuah *goal* yang sudah terpenuhi. Lacakbalik merupakan salah satu cara untuk melakukan proses perulangan.

* 1. Rekursi

Cara lain untuk melakukan perulangan adalah melalui rekursi. Prosedur rekursi adalah prosedur yang di dalamnya ada pemanggilan terhadap dirinya sendiri. Prosedur rekursi dapat merekam perkembangannya karena ia melewatkan (*passing*) pencacah, total, dan hasil sementara sebagai argumen dari satu iterasi ke iterasi berikutnya.

1. **List**

Pada Prolog, yang dimaksud dengan *list* adalah sebuah *object* yang didalamnya mengandung sejumlah *object* yang lain (jumlahnya dapat berubah-ubah). *List* dalam bahasa pemrograman lain bisa disamakan dengan tipe data *pointer* (C dan Pascal).

*List* adalah suatu data *object* jamak rekursif (*recursive compound object*). *List* terdiri dari 2 bagian yaitu *head*, yang merupakan elemen pertama dari *list* dan *tail*, elemen sisanya. *Tail* dari *list* adalah juga merupakan sebuah *list*, sedangkan *head* dari *list* merupakan sebuah elemen.

1. **Section Facts**

*Section facts* terdiri dari fakta-fakta yang mana fakta-fakta tersebut dapat ditambah dan dihapus secara langsung dari sebuah program pada saat program sedang berjalan (*at run time*). Kita dapat mendeklarasikan sebuah predikat pada *section facts* dan predikat tersebut dapat digunakan sama halnya seperti kalau dideklarasikan pada *section predicates*.

Kata kunci facts atau bisa juga database menandai permulaan sederetan deklarasi dari predikat yang ada pada *section facts*. Kita dapat menambahkan fakta-fakta (bukan *rule*) pada suatu *section facts* dari keyboard pada saat *run time* dengan menggunakan asserta dan assertz atau memanggil predikat consult untuk mengambil fakta tambahan dari sebuah file.

1. **Tentang Game Serigala Domba**

Ada seorang petani yang membawa seekor kambing, seekor serigala, dan sekeranjang sayur. Mereka berada di tepi sebuah pulau dan ingin menyebrang ke pulau seberang. Di tepi pulau itu hanya ada sebuah perahu yang cukup untuk dua penumpang. Dengan catatan lain, bahwa kambing tidak boleh ditinggal berdua dengan serigala tanpa adanya pemuda karena serigala akan memakan kambing, kemudian kambing tidak boleh ditinggal bersama sayur, karena kambing akan memakan sayurnya. Kemudian, yang bisa menggunakan perahu hanyalah petani. Bagaimana caranya agar mereka semua bisa menyebrang ke pulau seberang dengan utuh?

Sebenarnya caranya cukup mudah, dari kalimat di atas, memungkinkan serigala ditinggal bersama sayur. Jadi, yang pertama adalah petani menyebrangkan dirinya bersama kambing, dan biarkan serigala dan sayur ditinggal.

Kemudian tinggalkan kambing di pulau seberang, sementara petani kembali untuk membawa angkutan berikutnya. Seberangkan salah satu diantara serigala dan sayur, karena tidak akan ada masalah. Misalkan saja, petani menyebrangkan dirinya bersama sayur ke seberang.

Sesampainya diseberang, petani dan sayur turun. Karena apabila sayur ditinggal bersama kambing, kambing akan memakan sayur, maka kambing dibawa kembali ke pulau sebelumnya, sehingga sayur hanya sendirian.

Setelah membawa kambing ke pulau sebelumnya, berarti di pulau sebelumnya ada kambing, serigala, dan petani. Apabila petani membawa kambing, maka kambing tidak bisa ditinggal bersama sayur, jadi petani membawa serigala untuk diseberangkan ke pulau seberang.

Setelah serigala diseberangkan, tinggalkan serigala bersama sayur, itu akan aman-aman saja. Setelah itu, petani menjemput kambing yang masih berada di pulau sebelumnya. Dan akhirnya semuanya telah menyeberang.

Jika dibuat secara point-point maka alur logikanya seperti berikut :

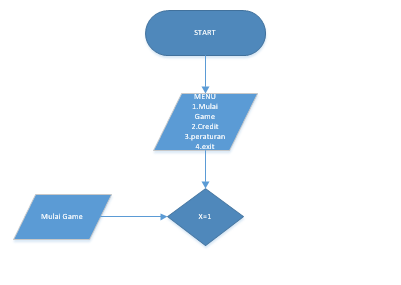
1. Petani menyebrang bersama Angsa dengan menaiki perahu, dan disebrang tinggal Serigala dan Padi.

1. Petani menurunkan Angsa di sebrang sungai dan Petani kembali ke perahu untuk menyebrangi sungai kembali.
2. Petani membawa Serigala untuk ke sebrang sungai dengan menaiki perahu. Dimana Petani tersebut harus turun terlebih dahulu ke seberang sungai agar Angsa tidak dimakan oleh Serigala, jika Angsa diturunkan terlebih dahulu maka Angsa tersebut akan dimakan oleh Serigala, dan sebrang sungai hanya ada seekor Serigala.
3. Petani tersebut membawa kembali Angsa ke perahu untuk ke sebrang sungai dengan menaiki perahu. Dimana Petani tersebut harus turun terlebih dahulu ke sebrang sungai agar Angsa tidak memakan Padi tersebut, jika Angsa diturunkan terlebih dahulu maka Angsa akan memakan Padi tersebut.
4. Petani menyebrang sungai bersama Padi dengan menaiki perahu, dan disebrang hanya ada seekor Angsa.
5. Petani tersebut menurunkan Padi ke sebrang sungai, dan Serigala tersebut ditemani oleh Padi. Maka Serigala dan Padi tidak akan terjadi makan memakan yang ada disebelahnya.
6. Petani tersebut kembali ke sebrang sungai dengan menaiki perahu. Kemudian Petani tersebut membawa Angsa ke perahu untuk menyebrangi sungai tersebut dengan perahu.
7. Sesampainya di sebrang sungai Petani yang harus turun terlebih dahulu agar Padi tidak dimakan oleh Angsa dan Angsa tidak dimakan oleh Serigala.
8. Selesai.

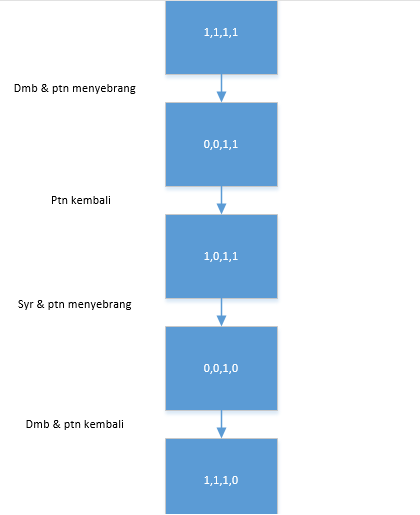
**BAB III**

**METODOLOGI**

**3.1 Alur Pembuatan Sistem**

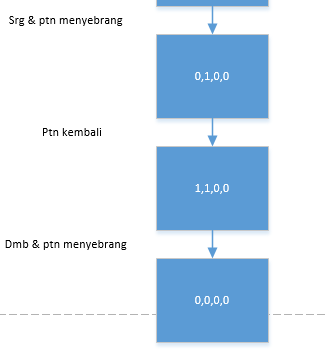
  
Gambar 3.1. Alur pertama

Tahapan pertama mulai dari start,selanjutnya akan tampil menu yang berisi mulai game,credit,peraturan,exit pertama kita pilih nomor 1 dan akan mulai bermain game



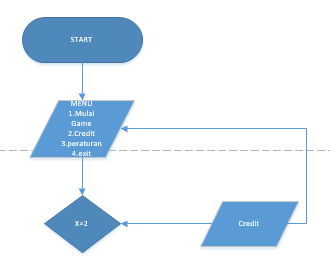
Gambar 3.2 Urutan permainan

Pertaman domba dan petani menyebrang,kemudian petani kembali lalu sayur dan petani menyebrang dan petani membawa domba kembali



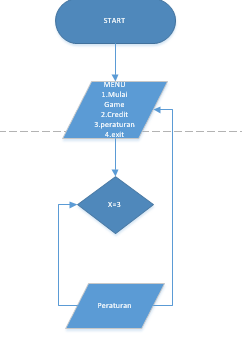
Gambar 3.3 Urutan permainan

Kemudian serigala menyebrang bersama petani dan petani kembali sendirian dan domba menyebrang bersama petani,dan serigala,petani,domba,dan sayuran sudah menyebrang ke kota b



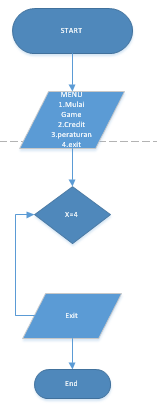
Gambar 3.4 Credit

Apabila kita memasukkan 2 kita akan meilhat kredit berisi tentang game tersebut dan di susun oleh siapa saja



Gambar 3.5 Peraturan

Ketika kita memasukkan 3 maka akan muncul peraturan cara bermain dimana program hanya membaca tipe integer,Perahu hanya memuat maksimal dua penumpang,Serigala tidak boleh ditinggal bersama domba,domba tidak boleh ditinggal bersama sayuran,dan perahu hanya boleh berangkat jika ada petani

  
Gambar 3.6 Exit

Ketika kita memasukkan 4 maka kita akan exit dari permainan serigala domba ini

**BAB IV**

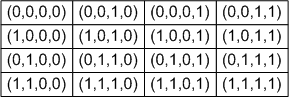
**HASIL DAN PEMBAHASAN**

4.1 Tabel Kebenaran dan Pohon Keputusan

4.1.1 Tabel Kebenaran

|  |  |
| --- | --- |
| Posisi | Aturan yang dipakai |
| (0, 0, 0, 0) | 1 |
| (1, 0, 1, 0) | 7 |
| (0, 0, 1, 0) | 3 |
| (1, 0, 1, 1) | 4 |
| (0, 0, 0, 1) | 2 |
| (1, 1, 0, 1) | 7 |
| (0, 1, 0, 1) | 1 |
| (1, 1, 1, 1) | solusi |

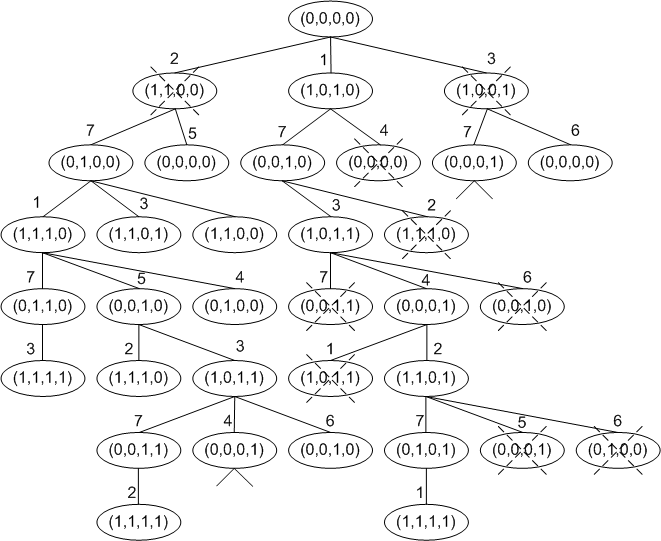
4.1.1.1 Tabel Kebenaran (1)



4.1.1.2 Tabel Kebenaran (2)

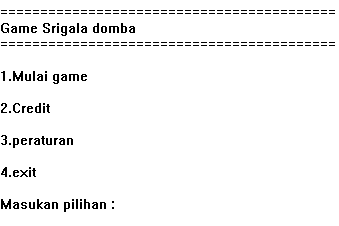
|  |  |
| --- | --- |
| **Aturan ke-** | **Aturan** |
| 1. | Petani dan Kambing menyeberang |
| 2. | Petani dan Sayuran menyeberang |
| 3. | Petani dan Serigala menyeberang |
| 4. | Petani dan Kambing kembali |
| 5. | Petani dan Sayuran kembali |
| 6. | Petani dan Serigala kembali |
| 7. | Petani kembali |

4.1.1.3 Tabel Kebenaran (3)

4.1.2 Pohon Keputusan

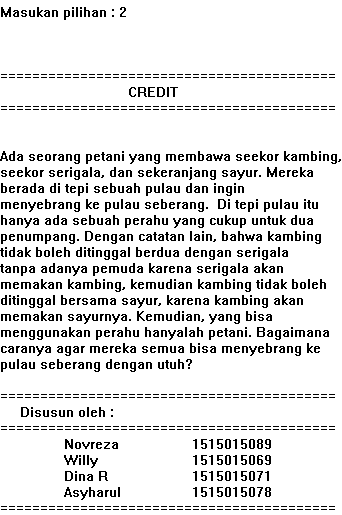
4.1.2.1 Pohon Keputusan

4.2 Analisis Program



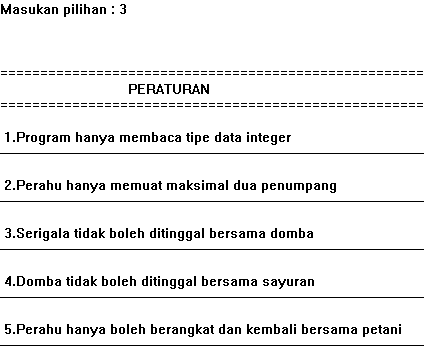
Gambar 4.2.1 Tampilan Awal

Ditampilan pertama program akan menampilkan beberapa pilihan yang diantaranya ada Mulai game, Credit, Peraturan, dan Exit. Jika ingin melanjutkan maka pilih salah satu nomor diatas (1-4).



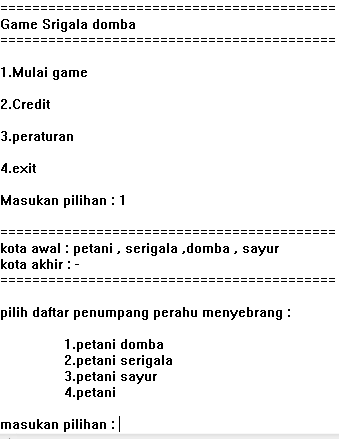
Gambar 4.2.2 Tampilan Credit

Selanjutnya kita pilih option ke-2 yaitu Credit, didalam credit terdapat cerita alur game Serigala Domba dan nama-nama anggota kelompok.



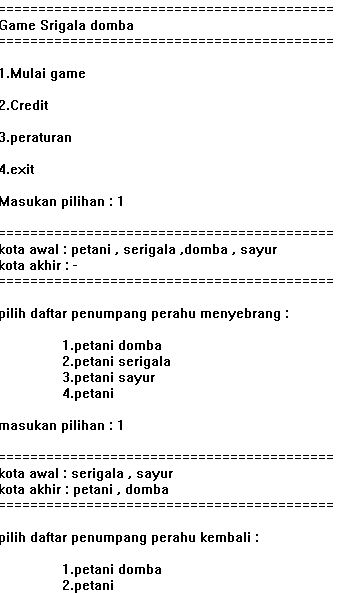
Gambar 4.2.3 Tampilan Peraturan

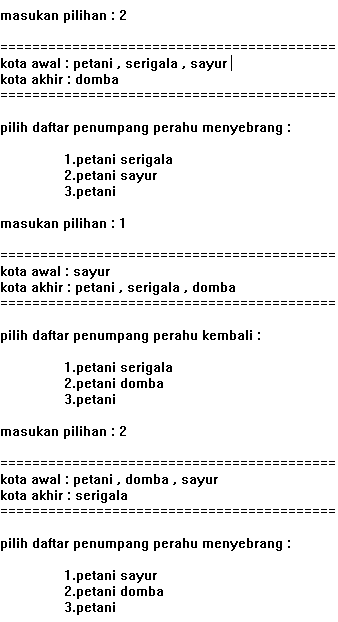
Dipilihan ini akan menampilkan peraturan dari game Serigala Domba, agar pengguna mengerti cara bermain game ini.



Gambar 4.2.4 Tampilan Mulai Game

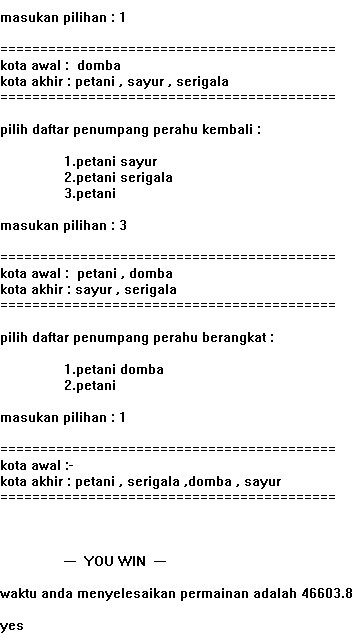
Pada saat memulai game maka akan muncul tampilan seperti diatas, terdapat kota awal dan kota akhir. Kota awal adalah dimana petani, serigala, domba, sayur berada dan petani ingin menyeberangkan semua bawaannya ke kota akhir. Program akan menampilkan 4 pilihan yang akan dipilih oleh pengguna nantinya.





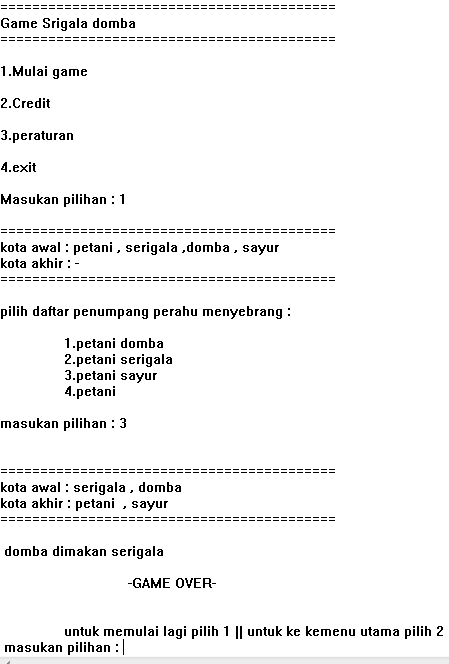
Gambar 4.2.5 Tampilan Game Berjalan

Jika pengguna menekan 1 maka pilihan nomer 1 yaitu petani dan domba akan berpindah ke kota akhir, karena yang dipilih akan menaiki perahu dan menyeberang. Selanjutnya program akan menampilkan pilihan berikutnya yaitu “pilih daftar penumpang perahu kembali”, karena perahu harus mengambil bawaan lainnya di kota awal maka perahu harus kembali dan didalam perahu wajib ada petani untuk mengendalikan perahunya, jadi program akan menampilkan 2 pilihan yaitu petani dan domba serta petani saja. Jika tekan 2 maka petani akan berpindah ke kota awal dan domba tetap berada di kota akhir. Program akan terus berlanjut hingga semua bawaan berada di kota akhir atau hingga game over.



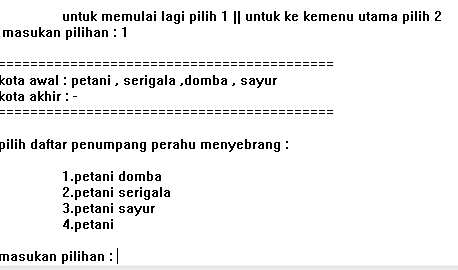
Gambar 4.2.6 Tampilan Game Win

Jika pengguna berhasil menyelesaikan game dengan tepat dan menyeberangkan semua bawaan petani ke kota akhir maka program akan berhenti dan menampilkan hasil serta waktu anda bermain.



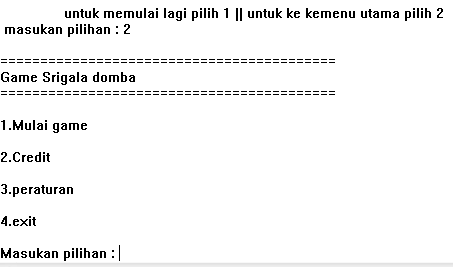
Gambar 4.2.7 Tampilan Game Over

Apabila pengguna salah atau gagal memasukkan jawaban maka program akan menampilkan alasan mengapa dia salah memilih jawaban tersebut dan program akan menampilkan pilihan lagi dibawah jika ingin bermain lagi atau ingin kembali ke menu utama.



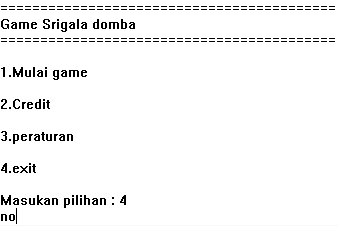
Gambar 4.2.8 Tampilan Bermain lagi

Jika pengguna menekan 1 maka program akan menampilkan lagi game dari awal.



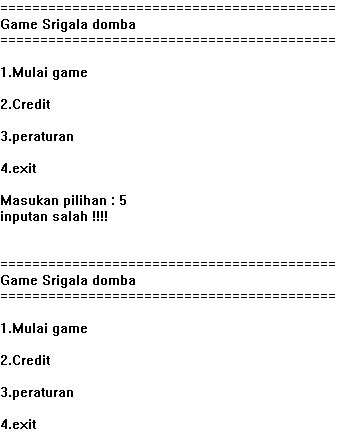
Gambar 4.2.9 Tampilan kembali ke Menu Utama

Jika pengguna menekan 2 maka program akan kembali ke menu utama.



Gambar 4.2.10 Tampilan Exit

Pada menu utama, jika menekan angka 4 maka kita telah keluar dari program.



Gambar 4.2.11 Tampilan Perulangan Pada Menu Utama

Pada menu utama, jika kita tekan tombol selain angka 1-4, maka akan mengulang ke menu utama dan muncul peringatan kalau inputan salah hingga menekan angka 1-4.

**BAB V**

**PENUTUP**

5.1 Kesimpulan

Kesimpulan dari pembuatan game Serigala Domba ini adalah bahwa game ini dibuat untuk mengasah otak dan game ini sangat bermanfaat untuk dimainkan karena game ini membutuhkan logika untuk memainkan dan menyelesaikannya.

5.2 Saran

Demikian Laporan Akhir yang dapat penulis paparkan, tentunya masih banyak kekurangan dan kelemahannya, karena keterbatasan pengetahuan, referensi serta kapabilitas penulis maka diharapkan pembaca dapat memberikan kritik dan saran yang bersifat membangun guna memperbaiki Lapoan Akhir ini dikesempatan-kesempatan berikutnya.

**DAFTAR PUSTAKA**

http://screamfreedome01.blogspot.co.id/2014/10/game-logika-algoritma.html

**LAMPIRAN**

**1. SYNTAX DATABASE**

domains

nama=string

facts

nondeterm daftarperaturan(nama)

nondeterm tulis(nama)

clauses

daftarperaturan(" 1.Program hanya membaca tipe data integer (bilangan bulat)").

daftarperaturan("\n 2.Perahu hanya memuat maksimal dua penumpang ").

daftarperaturan("\n 3.Serigala tidak boleh ditinggal bersama domba").

daftarperaturan("\n 4.Domba tidak boleh ditinggal bersama sayuran").

daftarperaturan("\n 5.Perahu hanya boleh berangkat dan kembali bersama petani").

tulis("\nAda seorang petani yang membawa seekor kambing, ").

tulis("\nseekor serigala, dan sekeranjang sayur. Mereka ").

tulis("\nberada di tepi sebuah pulau dan ingin").

tulis("\nmenyebrang ke pulau seberang. Di tepi pulau itu").

tulis("\nhanya ada sebuah perahu yang cukup untuk dua ").

tulis("\npenumpang. Dengan catatan lain, bahwa kambing").

tulis("\ntidak boleh ditinggal berdua dengan serigala ").

tulis("\ntanpa adanya pemuda karena serigala akan ").

tulis("\nmemakan kambing, kemudian kambing tidak boleh").

tulis("\nditinggal bersama sayur, karena kambing akan").

tulis("\nmemakan sayurnya. Kemudian, yang bisa ").

tulis("\nmenggunakan perahu hanyalah petani. Bagaimana").

tulis("\ncaranya agar mereka semua bisa menyebrang ke ").

tulis("\npulau seberang dengan utuh?").

tulis("\n\n==========================================\n Disusun oleh :\n==========================================\n\t").

goal

save("db.simpan").

**2. SYNTAX PROGRAM**

domains

nomer=integer

ok = string\*

nama=string

facts

nondeterm daftarperaturan(nama)

nondeterm tulis(nama)

predicates

nondeterm mulai(nomer)

nondeterm awal

nondeterm credit(ok)

nondeterm eror

nondeterm peraturan

nondeterm menugame

nondeterm pilih(nomer)

nondeterm pilih1(nomer)

nondeterm pilih2(nomer)

nondeterm pilih21(nomer)

nondeterm pilih3(nomer)

nondeterm pilih4(nomer)

nondeterm pilih5(nomer)

nondeterm pilih6(nomer)

nondeterm menu

nondeterm mulaiulang(nomer)

nondeterm tentang

clauses

menu:-

menugame.

menugame:-

write("\n==========================================\nkota awal : petani , serigala ,domba , sayur\nkota akhir : -\n==========================================\n\n"),

write("pilih daftar penumpang perahu menyebrang :\n\n\t1.petani domba\n\t2.petani serigala\n\t3.petani sayur\n\t4.petani\n\nmasukan pilihan : "),

readint(A),pilih(A).

pilih(A):-

A=1,write("\n==========================================\nkota awal : serigala , sayur \nkota akhir : petani , domba \n==========================================\n\n"),

write("pilih daftar penumpang perahu kembali :\n\n\t1.petani domba \n\t2.petani \n\nmasukan pilihan : "),readint(B),pilih1(B);

A=2,write("\n\n==========================================\nkota awal : sayur, domba \nkota akhir : petani , serigala \n==========================================\n\n sayur habis dimakan domba \n\n\t\t-GAME OVER-\n==========================================\n"),

write("\n\n\n\tuntuk memulai lagi pilih 1 || untuk ke kemenu utama pilih 2\n masukan pilihan : "),readint(B),mulaiulang(B);

A=3,write("\n\n==========================================\nkota awal : serigala , domba \nkota akhir : petani , sayur \n==========================================\n\n domba dimakan serigala \n\n\t\t-GAME OVER-"),

write("\n\n\n\tuntuk memulai lagi pilih 1 || untuk ke kemenu utama pilih 2\n masukan pilihan : "),readint(B),mulaiulang(B);

A=4,write("\n\n==========================================\nkota awal : serigala , sayur , domba\nkota akhir : petani \n==========================================\n\n domba dimakan serigala \n\n\t\t-GAME OVER-"),

write("\n\n\n\tuntuk memulai lagi pilih 1 || untuk ke kemenu utama pilih 2\n masukan pilihan : "),readint(B),mulaiulang(B);

A<>1,A<>2,A<>3,A<>4,write("\ninputan salah !!!!\n\n"),menugame.

pilih1(A):-

A=1,menugame;

A=2,write("\n==========================================\nkota awal : petani , serigala , sayur \nkota akhir : domba \n==========================================\n\n"),

write("pilih daftar penumpang perahu menyebrang :\n\n\t1.petani serigala\n\t2.petani sayur\n\t3.petani\n\nmasukan pilihan : "),

readint(B),pilih2(B);

A<>1,A<>2,write("\ninputan salah !!!!\n\n"),pilih(1).

pilih2(A):-

A=1,write("\n==========================================\nkota awal : sayur \nkota akhir : petani , serigala , domba \n==========================================\n\n"),

write("pilih daftar penumpang perahu kembali :\n\n\t1.petani serigala\n\t2.petani domba\n\t3.petani\n\nmasukan pilihan : "),

readint(B),pilih3(B);

A=2,write("\n==========================================\nkota awal : serigala \nkota akhir : petani , sayur , domba \n==========================================\n\n"),

write("pilih daftar penumpang perahu kembali :\n\n\t1.petani sayur\n\t2.petani domba\n\t3.petani\n\nmasukan pilihan : "),readint(B),pilih21(B);

A=3,pilih(1);

A<>1,A<>2,A<>3,write("\ninputan salah !!!!\n\n"),pilih1(2).

pilih21(A):-

A=1,pilih1(2);

A=2,write("\n==========================================\nkota awal : petani , serigala , domba \nkota akhir : sayur \n==========================================\n\n"),

write("pilih daftar penumpang perahu kembali :\n\n\t1.petani serigala\n\t2.petani domba\n\t3.petani\n\nmasukan pilihan : "),pilih4(1);

A=3,write("\n==========================================\nkota akhir : sayur, domba \nkota awal : petani , serigala \n==========================================\n\n sayur habis dimakan domba \n\n\t\t-GAME OVER-"),

write("\n\n\n\tuntuk memulai lagi pilih 1 || untuk ke kemenu utama pilih 2\n masukan pilihan : "),readint(B),mulaiulang(B);

A<>1,A<>2,A<>3,write("\ninputan salah !!!!\n\n"),pilih2(2)

pilih3(A):-

A=1,pilih1(2);

A=2,write("\n==========================================\nkota awal : petani , domba , sayur \nkota akhir : serigala \n==========================================\n\n"),

write("pilih daftar penumpang perahu menyebrang :\n\n\t1.petani sayur\n\t2.petani domba\n\t3.petani\n\nmasukan pilihan : "),

readint(B),pilih4(B);

A=3,write("\n==========================================\nkota awal : petani , sayur \nkota akhir : serigala , domba \n==========================================\n\n domba dimakan serigala \n\n\t\t-GAME OVER-"),

write("\n\n\n\tuntuk memulai lagi pilih 1 || untuk ke kemenu utama pilih 2\n masukan pilihan : "),readint(B),mulaiulang(B);

A<>1,A<>2,A<>3,write("\ninputan salah !!!!\n\n"),pilih2(1).

A=3,write("\n==========================================\nkota awal : petani , sayur \nkota akhir : serigala , domba \n==========================================\n\n domba dimakan serigala \n\n\t\t-GAME OVER-"),

write("\n\n\n\tuntuk memulai lagi pilih 1 || untuk ke kemenu utama pilih 2\n masukan pilihan : "),readint(B),mulaiulang(B);

A<>1,A<>2,A<>3,write("\ninputan salah !!!!\n\n"),pilih2(1).

pilih4(A):-

A=1,write("\n==========================================\nkota awal : domba \nkota akhir : petani , sayur , serigala \n==========================================\n\n"),

write("pilih daftar penumpang perahu kembali :\n\n\t1.petani sayur\n\t2.petani serigala\n\t3.petani\n\nmasukan pilihan : "),readint(B),pilih5(B);

A=2,pilih2(1);

A=3,write("\n==========================================\nkota awal : sayur, domba \nkota akhir : petani , serigala \n==========================================\n\n sayur habis dimakan domba \n\n\t\t-GAME OVER-"),

write("\n\n\n\tuntuk memulai lagi pilih 1 || untuk ke kemenu utama pilih 2\n masukan pilihan : "),readint(B),mulaiulang(B);

A<>1,A<>2,A<>3,write("\ninputan salah !!!!\n\n"),pilih3(2).

pilih5(A):-

A=1,pilih3(2);

A=2,pilih21(2);

A=3,write("\n==========================================\nkota awal : petani , domba \nkota akhir : sayur , serigala \n==========================================\n\n"),

write("pilih daftar penumpang perahu berangkat :\n\n\t1.petani domba\n\t2.petani \n\nmasukan pilihan : "),readint(B),pilih6(B);

A<>1,A<>2,A<>3,write("\ninputan salah !!!!\n\n"),pilih4(1).

pilih6(A):-

A=1,write("\n==========================================\nkota awal :- \nkota akhir : petani , serigala ,domba , sayur\n==========================================\n\n\n\n\t--- YOU WIN ---"),nl,nl;

A=2,pilih4(1);

A<>1,A<>2,write("\ninputan salah !!!!\n\n"),pilih5(3).

mulaiulang(X):-

X=1,menu;

X=2,awal.

mulai(X):-

X=1,marktime(0,T1),

menu,marktime(0,T2),

difftime(T2,T1,Time)

,write("waktu anda menyelesaikan permainan adalah ",Time),nl,nl;

X=2,tentang,credit(["Novreza\t\t1515015089","\tWilly\t\t1515015069","\tDina R\t\t1515015071","\tAsyharul\t\t1515015078\n=========================================="]),awal;

X=3,peraturan,awal;

X=4,!;

X<>1,X<>2,X<>3,X<>4,eror.

eror:- write("inputan salah !!!!\n\n"),awal.

credit([]).

credit([H|T]):-

write(H),nl,

credit(T).

awal:- write("\n==========================================\nGame Srigala domba\n==========================================\n\n1.Mulai game\n\n2.Credit\n\n3.peraturan\n\n4.exit\n\nMasukan pilihan : "),readint(Pilih),mulai(Pilih).

tentang:-write("\n\n\n==========================================\n\t\tCREDIT\n==========================================\n"),nl,tulis(Y),

write(Y),fail.

tentang.

peraturan:- write("\n\n\n=====================================================\n\t\tPERATURAN\n=====================================================\n"),nl,daftarperaturan(Y),

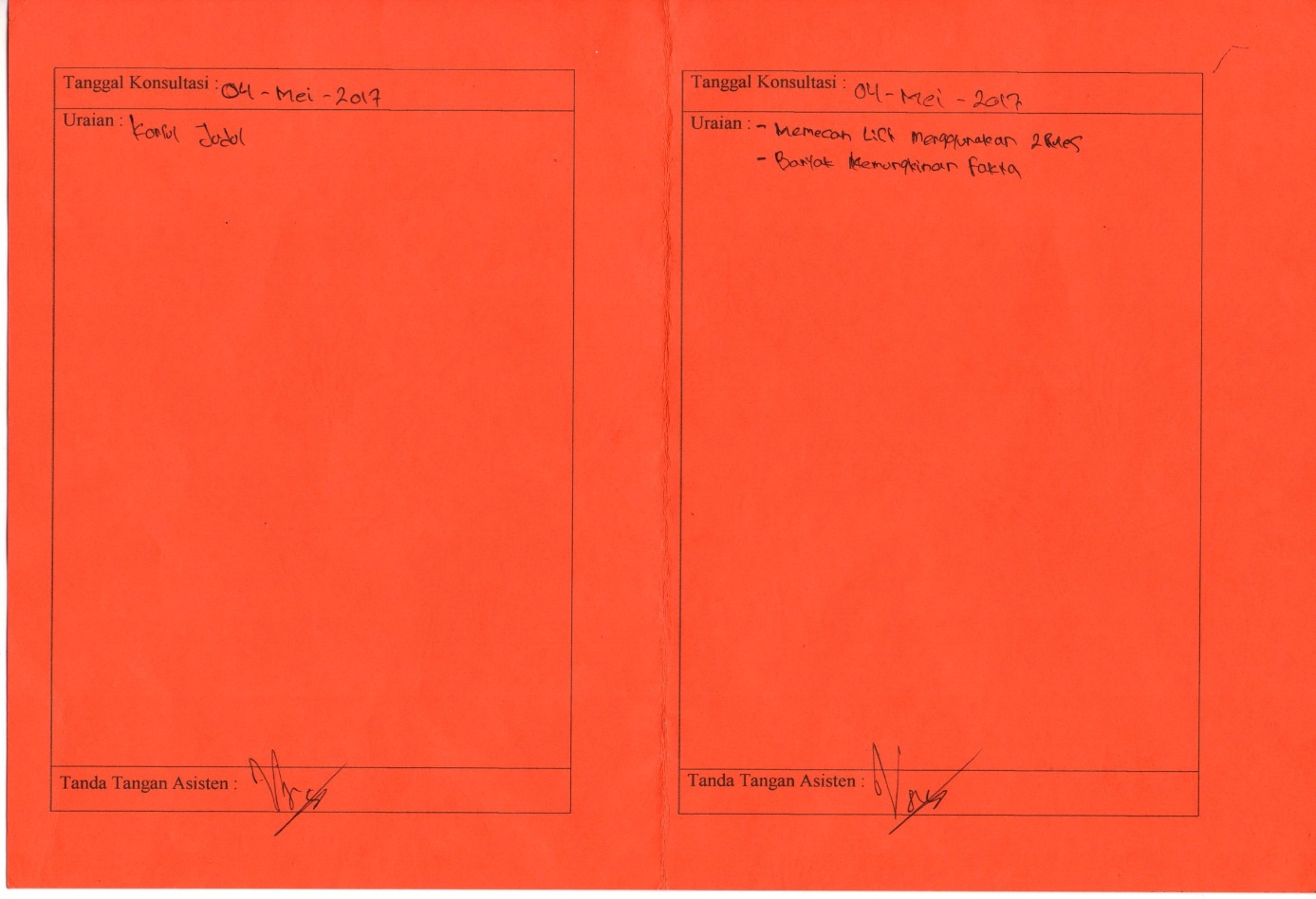
write(Y,"\n----------------------------------------------------------------------------------------------------------"),nl,fail.

peraturan.

goal

consult("db.simpan"),

awal.

**LAMPIRAN**