

Congklak
Minimax Algorithm and Alpha Beta Pruning

LAPORAN TUGAS BESAR I IF3170

Inteligensi Buatan



Disusun oleh :

Lukas Kurnia Jonathan	13517006
Eginata Kasan	13517030
Vivianni	13517060
Rika Dewi	13517147

PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
BANDUNG

2019

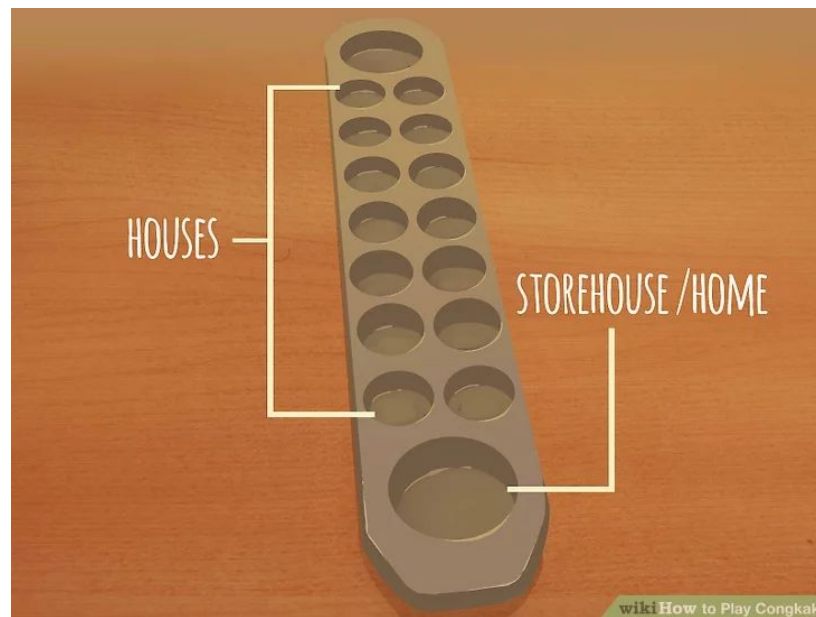
A. Congklak

Congklak adalah salah satu permainan yang memiliki karakteristik: deterministic, fully observable, static, discrete, sequential, turn-taking, two-players. Pada tugas besar kali ini, kami memilih permainan congklak untuk dieksplorasi menggunakan bahasa pemrograman Python3.

B. Game Play

Pada permainan congklak, terdapat *house* dan *storehouse* yang di dalamnya terdapat *shell*. *Shell* adalah biji yang akan digunakan sebagai poin pada permainan congklak.

Masing-masing pemain memiliki 7 buah *house* yang berisi 7 buah *shell*.



Gambar 1. *Store* dan *Storehouse*

Objektif dari permainan ini adalah mengumpulkan *shell* sebanyak-banyaknya di *storehouse*. Permainan akan berakhir ketika semua *house* sudah tidak memiliki *shell* yang tersisa. Pemenang adalah pemain dengan jumlah *shell* terbanyak pada *storehouse*.



Gambar 2. *Storehouse*

Pemain pertama akan mengambil *shell* dari salah satu *house* daerahnya dan berjalan searah jarum jam menaruh satu buah *shell* di masing-masing *house* yang dilewati. Ketika melewati *storehouse* lawan, pemain tidak menaruh *shell* di tempat tersebut.



Gambar 3. Proses memindahkan *shell*

Beberapa kondisi yang mungkin terjadi ketika menaruh *shell* terakhir pada sebuah *house*:

1. *Shell* terakhir jatuh di *storehouse* sendiri, maka pemain mendapat giliran kembali untuk memainkan permainan
2. *Shell* terakhir terjatuh di *house* sendiri
 - a. Jika sudah melakukan sekali *loop* dan *house* lawan di seberang memiliki sejumlah *shell*, maka ambil *shell* lawan dan *shell* pemain kemudian ditambahkan ke *storehouse* pemain.
 - b. Jika tidak terdapat *shell* di *house* seberang maka giliran berakhir.
3. *Shell* terakhir jatuh di *house* lawan, maka giliran berakhir



Gambar 4. Merebut *shell* lawan dan memasukkan ke dalam *storehouse* pemain

Cara menjalankan aplikasi:

1. Pertama-tama harus disiapkan dulu beberapa kebutuhan untuk menjalankan aplikasi. Diantaranya python3 dan kivy (untuk menampilkan GUI)
2. Run python3 main.py pada terminal untuk menjalankan aplikasi
3. Masukan mode permainan, terdapat 3 mode permainan yaitu:
 - Bot Minimax vs Player
 - Bot Random vs Player
 - Bot Minimax cs Random
4. Jika mode permainan yang dipilih adalah Bot Minimax vs Player atau Bot Minimax vs Bot Random, pilih tingkat kesulitan (Easy/Hard)
5. Pilih giliran pertama pemain
6. Jika Anda bermain sebagai player, tekan angka pada board congklak saat giliran Anda
7. Lakukan kembali langkah 6 hingga permainan selesai

C. Tahapan Pembangunan Aplikasi

Beberapa langkah yang harus dikerjakan untuk membangun aplikasi congklak dengan MiniMax:

- Membuat representasi *state*, mendefinisikan *rule* permainan congklak pada bahasa pemrograman python3
- Menetapkan fungsi evaluasi
 - Fungsi evaluasi didapatkan dari mengurangi nilai house pemain terhadap house lawan
- Membuat algoritma pencarian nilai terbaik dengan MiniMax Algorithm dan Alpha Be
- Bot move: 10ta Pruning untuk mengetahui langkah mana yang akan dipilih
- Membuat GUI dari program

D. Representasi proses pada aplikasi permainan

- State
 - State yang didefinisikan pada aplikasi kami untuk merepresentasikan kondisi pada suatu saat adalah menggunakan sebuah struktur data *array of integer* dengan ukuran 16.
 - Representasi state pada board congklak:
[8] [9] [10] [11] [12] [13] [14] [15]
[7] [6] [5] [4] [3] [2] [1] [0]

Keterangan:

- Indeks ke-0 sampai ke-6 adalah *house* pemain
- Indeks ke-8 sampai ke-14 adalah *house* lawan
- Indeks ke-7 adalah *storehouse* pemain
- Indeks ke-15 adalah *storehouse* lawan
- Giliran bergerak
 - Giliran bergerak pada aplikasi kami direpresentasikan dengan konstanta NORTH_TURN dengan nilai 1 dan SOUTH_TURN dengan nilai 0.
 - NORTH_TURN memiliki wilayah dari indeks ke-8 hingga ke-15
 - SOUTH_TURN memiliki wilayah dari indeks ke-0 hingga ke-7
- Legal move
 - Legal move pada setiap aksi terdapat pada setiap indeks yang merupakan *house*.
 - Beberapa *constraint* yang harus diperhatikan:
 - Saat NORTH_TURN
 - Indeks yang dipilih hanya terdapat pada range indeks-8 hingga ke-14
 - Jumlah *shell* pada indeks yang dipilih tidak boleh 0
 - Jumlah *shell* pada seluruh daerah northSite tidak boleh 0
 - Saat SOUTH_TURN

- Indeks yang dipilih hanya terdapat pada range indeks-0 hingga ke-6
- Jumlah *shell* pada indeks yang dipilih tidak boleh 0
- Jumlah *shell* pada seluruh daerah southSite tidak boleh 0
- Fungsi evaluasi
 - Fungsi evaluasi yang digunakan adalah house dari board pemain berada sekarang dikurang dengan house dari board lawan

E. Proses Pencarian dengan Algoritma Minimax dan Alpha Beta Pruning

- Algoritma MinimaxgetSouthStoreHousedilakukan. Algoritma ini akan menghasilkan semua kemungkinan yang bisa dijalankan dari root. Kemudian dari semua kemungkinan yang ada, akan dijalankan fungsi algoritma minimax. Pada algoritma minimax, jika kondisi yang akan bermain berupa giliran board, maka yang dijalankan adalah algoritma maksimum yang akan mengembalikan nilai evaluasi maksimum dari state dengan kedalaman yang sama. Jika kondisi sekarang adalah giliran lawan yang bermain, maka akan dijalankan fungsi minimum yang akan menghasilkan nilai evaluasi yang memberikan kerugian minimum
- Algoritma Alpha Beta Pruning
Algoritma Alpha Beta Pruning pada program congklak ini digunakan pada algoritma minimax untuk mengoptimisasi pencarian yaitu dengan cara mengurangi jalur pencarian yang tidak diperlukan. Algoritma ini akan mempersingkat waktu pencarian yang dibutuhkan.

F. Data pertandingan permainan congklak

1. Bot Minimax vs Manusia

Easy Mode

	Bot Minimax	Manusia
Win	5	5

Hard Mode

	Bot Minimax	Manusia
Win	9	1

2. Bot Random vs Manusia

	Bot Random	Manusia
--	------------	---------

Win	2	8
-----	---	---

3. Bot Minimax vs Bot Random

Easy Mode

	Bot Minimax	Bot Random
Win	5	5

Hard Mode

	Bot Minimax	Bot Random
Win	9	1

Referensi

<https://kivy.org/doc/stable/gettingstarted/intro.html#>, diakses pada 29 September 2019

<https://www.wikihow.com/Play-Congkak>, diakses pada 30 September 2019

<https://www.geeksforgeeks.org/minimax-algorithm-in-game-theory-set-4-alpha-beta-pruning/>, diakses pada 30 September 2019