

Laporan Tugas Besar I

IF3170 Inteligensi Buatan

Minimax Algorithm and Alpha Beta Pruning in Adjacency Strategy Game



Disusun Oleh:

Matthew Mahendra	13521007
Muhamad Salman Hakim Alfarisi	13521010
Eunice Sarah Siregar	13521013
Syarifa Dwi Purnamasari	13521018

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BANDUNG

2023

Daftar Isi

Objective Function	3
Minimax Algorithm and Alpha Beta Pruning	4
Local Search	6
Genetic Minimax Algorithm	7
Pengujian	8

Objective Function

Fungsi objektif (f) yang digunakan adalah menggunakan perhitungan,

$$f(i) = \text{countMarkaPemain}(i) - \text{countMarkaLawan}(i)$$

dengan i adalah semua state neighbor dari current state dan i dimulai dari -1 dengan indeks -1 adalah current state.

Fungsi mencari selisih jumlah marka pemain (“X” atau “O”) dengan marka lawan (“O” atau “X”). Solusi optimal didapatkan jika $f(i)$ memberikan selisih terbesar dan nilainya tidak negatif. Dengan demikian, dapat diambil kesimpulan bahwa jumlah marka pemain sudah melebihi marka lawan.

Minimax Algorithm and Alpha Beta Pruning

Minimax algorithm merupakan metode pengambilan keputusan yang digunakan dalam *artificial intelligence*, *decision making*, *game theory*, *statistics*, dan *philosophy* untuk meminimalkan kerugian yang mungkin terjadi dalam skenario terburuk (kerugian maksimum). Dalam konteks tugas besar ini, minimax algorithm digunakan untuk menentukan langkah terbaik bagi seorang pemain dengan anggapan bahwa pemain lawan juga bermain secara optimal. Algoritma ini umumnya digunakan dalam *turn-based games* dengan 2 pemain seperti Tic-Tac-Toe, Backgammon, Chess, dan lain-lain. Pada minimax algorithm terdapat 2 fungsi utama, yaitu maximizer dan minimizer. Pada maximizer, pemain berusaha untuk mencapai skor tertinggi yang memungkinkan, sementara pada minimizer, pemain berusaha untuk melakukan langkah sebaliknya yaitu meminimalkan skor atau nilai yang mungkin dicapai. Algoritma minimax memiliki salah satu bentuk optimasi yaitu Alpha-Beta Pruning. Pada Minimax Alpha-Beta Pruning, dilakukan eliminasi bagian dari pohon yang tidak mempengaruhi keputusan akhir.

Proses pencarian minimax bergantung pada *root* yang sedang aktif. *Root* yang sedang aktif merupakan *current state game*. Ketika *current state (root node)* adalah giliran pemain maka akan dilakukan pencarian nilai minimal dari *child node* (giliran lawan). Pada *child node* dilakukan pencarian nilai maksimal dari *child node* dibawahnya (giliran pemain). Proses ini berlangsung secara rekursif berdasarkan *current state*. Pada permainan ini, node berisi jumlah marka dari pemain. Bot yang dibuat, dirancang untuk mendapatkan jumlah marka paling banyak pada papan dari kemungkinan yang dimaksimalkan setiap giliran pemain. Akibat proses pencarian minimax menggunakan optimasi Alpha-Beta Pruning, di setiap proses pemaksimalan marka pemain disimpan nilai alpha (nilai tertinggi) untuk eliminasi *state* yang tidak diperlukan. Sebaliknya, ketika proses minimalisasi marka lawan disimpan nilai beta (nilai terendah). Nilai alpha-beta akan terus diupdate setiap cabang node.

Pengambilan gerakan oleh bot yang dibuat hanya dapat dilakukan pada tingkat 1 dari pohon permainan. Evaluasi dari minimax akan memberikan keputusan cabang manakah yang akan diambil sebagai gerakan berikut dari *state root*. Keputusan yang diambil adalah nilai evaluasi minimax tertinggi dari setiap cabang.

Pada proses pencarian minimax yang kami buat, terdapat batasan *depth* pada setiap pemanggilan fungsi minimax. tiap pemanggilan fungsi minimax. Batasan *depth* yang dibuat adalah maksimal 3 *depth*. Pembatasan *depth* ini akan mengurangi pembangunan pohon permainan, agar tidak membangkitkan terlalu banyak cabang. Hal ini dilakukan agar proses pencarian solusi tidak terlalu lama ketika jumlah ronde dalam pertandingan cukup banyak, namun tetap memberikan kesempatan untuk algoritma memikirkan aksi untuk 2 langkah berikutnya.

Local Search

Local search adalah algoritma yang pada umumnya digunakan untuk memberikan solusi terbaik untuk sebuah masalah dengan sebuah solusi random yang akan membawa perubahan kecil untuk mendapatkan solusi terbaik. Berbagai macam jenis algoritma local search. Pada konteks tugas besar ini, jenis local search yang digunakan adalah stochastic hill climbing. Algoritma ini memanfaatkan tidak mengidentifikasi semua tetangganya, tetapi melakukan pemilihan node random dan memutuskan apakah akan berpindah ke node tersebut atau memeriksa node yang lainnya.

Pada, stochastic Hill Climbing Search akan membangkitkan semua state tetangga dengan melakukan semua pergerakan yang memungkinkan dan disimpan sebagai current state. Selanjutnya dengan menggunakan fungsi random akan diambil satu state dari state-state yang sebelumnya berhasil dibangkitkan. Proses diulang sebanyak n -kali, dengan $n = 64$. Jika state tetangga lebih baik daripada state current, maka state tersebut diambil sebagai state current. Perbandingan menggunakan fungsi objektif yang didefinisikan.

Algoritma Stochastic Hill Climbing Search digunakan sebagai local search pada permasalahan ini, karena dapat melakukan perhitungan aksi yang cukup cepat serta hemat. Selain itu, pergerakan secara *random* memungkinkan untuk pergerakan yang lebih banyak kemungkinannya, dengan tetap memberikan state suksesor yang sesuai dengan fungsi objektif yang digunakan.

Genetic Minimax Algorithm

Genetic algorithm adalah metode optimasi untuk menyelesaikan masalah berdasarkan seleksi alam sesuai kriteria Darwin tentang evolusi populasi. Untuk bertahan hidup dari seleksi alam, dibutuhkan individu yang unggul dengan menggunakan fitness function sebagai gambaran kelayakan solusi terhadap masalah. Salah satu bentuk modifikasi dari genetic algorithm adalah genetic minimax algorithm. Genetic minimax algorithm menggunakan skema representasi kromosom yang diusulkan untuk menggambarkan keadaan permainan dua pemain. Algoritma ini menggunakan pohon reservasi untuk membantu dalam proses evaluasi.

Proses genetic minimax algorithm diawali dengan menentukan populasi awal yang berisikan kromosom. Selanjutnya, tiap kromosom akan dihitung fitness function dengan memanfaatkan heuristic. Hasil fitness function yang terbaik akan diambil empat buah yang nantinya akan menjadi parent. Setelah itu, parent yang didapatkan akan dilakukan crossover untuk mendapatkan parent yang baru. Dari parent hasil crossover tersebut, kemudian dilakukan mutasi pada parent yang dipilih secara random dengan value yang dipilih secara random juga dari populasi awal. Setelah didapatkan hasil parent yang baru, lakukan pengecekan validasi pada parent, apakah parent merupakan node kosong atau bukan. Jika parent valid, generate minimax pada parent yang berhasil dibentuk. Keputusan yang diambil adalah nilai evaluasi minimax tertinggi dari setiap cabang.

Pengujian

Pengujian Bot Minimax vs Manusia

Pengujian dilakukan sebanyak 5 permainan dengan jumlah ronde 28, 11, 10, 9, dan 8.

Untuk 28 Ronde didapatkan hasil sebagai berikut,

Game Board Display								— □ ×	
O	O	O	O	O	X	O	O	Number Of Rounds Left:	0
O	X	X	X	X	O	X	O		
X	X	X	O	O	X	O	X		
O	X	O	X	X	O	X	O	Player X	Player O
X	X	O	X	X	O	X	X	Human	Minimax (Winner!)
O	O	O	X	X	O	O	X	29	35
O	O	X	O	X	X	O	O	End Game Play New Game	
X	O	O	O	O	X	O	O		

Untuk 11 ronde didapatkan hasil sebagai berikut,

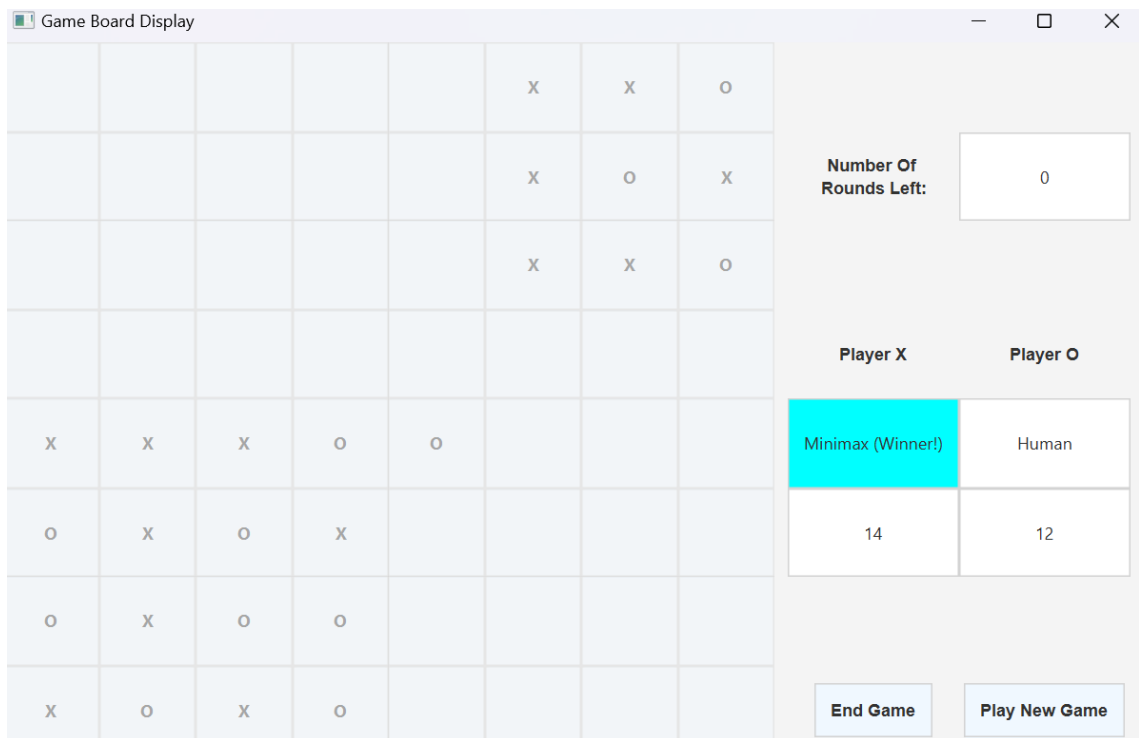
IF3170 Inteligensi Buatan
Tugas Besar I

Game Board Display									
X				X	X	X	O	Number Of Rounds Left:	0
	X			X	X	O	X		
O	O			X	O	O	X		
			O	X	O	X	X		
			X	O	O			Player X	Player O
			X	X				Minimax (Winner!)	Manusia
X	X							20	10
X	X							End Game	Play New Game

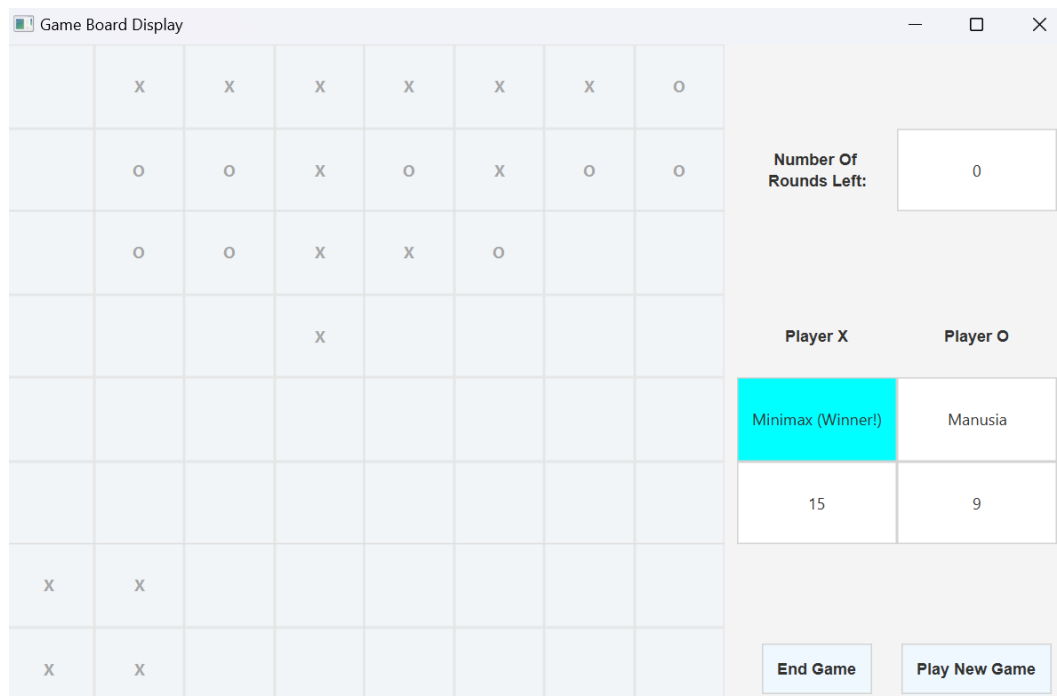
Untuk 10 ronde didapatkan hasil sebagai berikut,

Game Board Display									
O	X				X	X	O	Number Of Rounds Left:	0
X	X	X			X	X	X		
	O						X		
X	O	O							
X	O	O						Player X	Player O
O	O	O						Minimax (Winner!)	Manusia
O	X	X						17	11
X	X	X						End Game	Play New Game

Untuk 9 ronde didapatkan hasil sebagai berikut,



Untuk 8 Ronde didapatkan hasil sebagai berikut,



Kemenangan bot minimax 100%.

Berdasarkan hasil pertandingan manusia vs minimax, dapat dilihat bahwa persentase kemenangan bot 100%. Hal tersebut disebabkan adanya kecenderungan algoritma minimax untuk memaksimalkan nilai objective function. Selain itu, untuk mengoptimasinya juga menggunakan Alpha-Beta Pruning dengan melakukan eliminasi yang tidak mempengaruhi keputusan akhir.

Pengujian Bot Local Search vs Manusia

Pengujian dilakukan sebanyak 3 kali dengan jumlah ronde 28, 9, dan 8.

Untuk 28 ronde didapatkan hasil sebagai berikut,

Game Board Display									
O	O	X	O	O	O	X	O	Number Of Rounds Left:	0
O	X	X	O	O	X	O	O		
X	O	X	O	X	X	O	O		
X	O	X	O	O	O	O	X		
O	X	X	O	O	O	X	O	Player X	Player O
X	X	X	X	X	O	X	O	Stochastic	Manusia (Winner!)
O	X	O	X	O	X	X	O	29	35
X	O	X	O	X	X	O	O		
								End Game	Play New Game

Untuk 8 ronde didapatkan hasil sebagai berikut,

Game Board Display									
						O	O	Number Of Rounds Left:	0
		O		O	O	O	O		
			O	O					
X	X	X			X	X			
	X		O	X	X			Player X	Player O
		X	X			X	X	Local Search (Winner!)	Manusia
X	X							16	10
X	X								
								End Game	Play New Game

Untuk 9 ronde didapatkan hasil sebagai berikut,

Game Board Display									
			O	O	O	X	O		
			X	O	O	X	O	Number Of Rounds Left:	0
				O		X	X		
X			O	O		X		Player X	Player O
				O	O	O	X	Local Search	Manusia
								13	13
X	X					X			
X	X							End Game	Play New Game

Kemenangan bot local search 66.67% .

Berdasarkan hasil pertandingan local search vs manusia tersebut, didapatkan hasil yang tidak optimal seperti pada 28 ronde karena pada algoritma local search Stochastic Hill-Climbing Search yang digunakan, *neighbor node* dari *current node* yang diperiksa, dipilih secara acak sedangkan manusia sendiri dapat memilih value terbesar dengan melihat *current state* dari *board* permainan.

Pengujian Bot Minimax vs Bot Local Search

Pengujian dilakukan untuk 3 permainan dengan jumlah ronde 28, 9, dan 8 ronde.

Didapatkan untuk permainan 28 ronde,

Game Board Display

O	X	X	O	O	X	X	O
O	O	X	X	O	X	X	X
O	X	X	O	X	X	O	X
X	X	X	X	X	O	X	X
X	X	X	X	X	X	O	X
X	O	X	O	X	O	X	X
O	X	O	X	O	X	O	O
X	O	O	X	X	O	O	O

Number Of Rounds Left:

0

Player X

Player O

Minimax (Winner!)

Hill Climbing

39

25

End Game

Play New Game

Untuk permainan 9 ronde didapatkan hasil,

Game Board Display

X	X			X	O	X	O
		X	X	X	X	X	O
					X		
O	O		O				
X							
							X
X	O	O					X
X	X				X	X	

Number Of Rounds Left:

0

Player X

Player O

Minimax (Winner!)

Hill Climbing

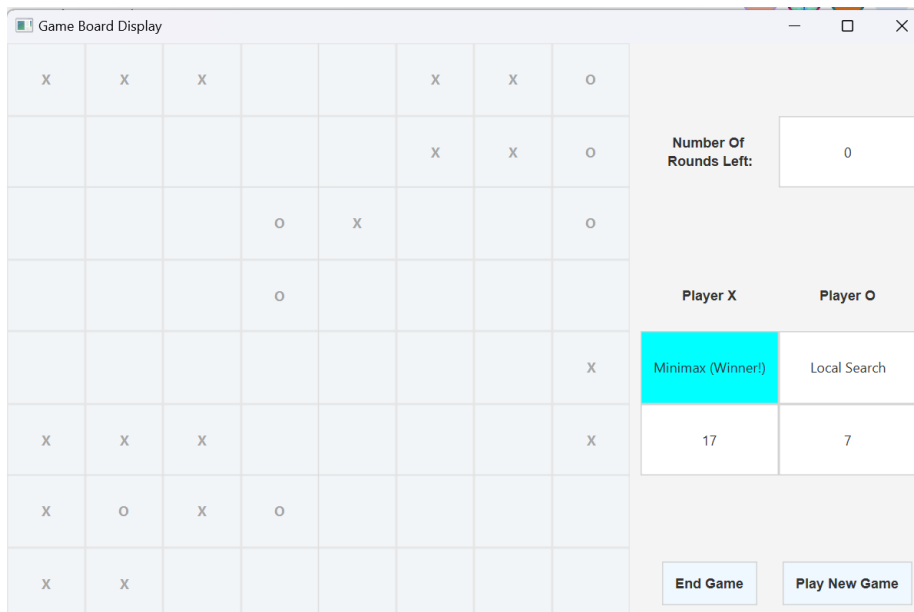
18

8

End Game

Play New Game

Untuk permainan 8 ronde didapatkan hasil,



Kemenangan bot Minimax 100% dan kemenangan bot local search 0%.

Berdasarkan pertandingan tersebut, dapat dilihat bahwa bot minimax memiliki persentase kemenangan lebih besar dibandingkan local search. Minimax memiliki persentase kemenangan 100% karena dapat memikirkan aksi untuk 2 langkah berikutnya. Sedangkan pada local search memiliki kecenderungan untuk random sehingga terdapat kemungkinan untuk menghasilkan hasil yang buruk.

Pengujian Bot Genetic vs Bot Local Search

Pengujian dilakukan untuk 3 permainan dengan jumlah ronde 28, 9, dan 8 ronde.

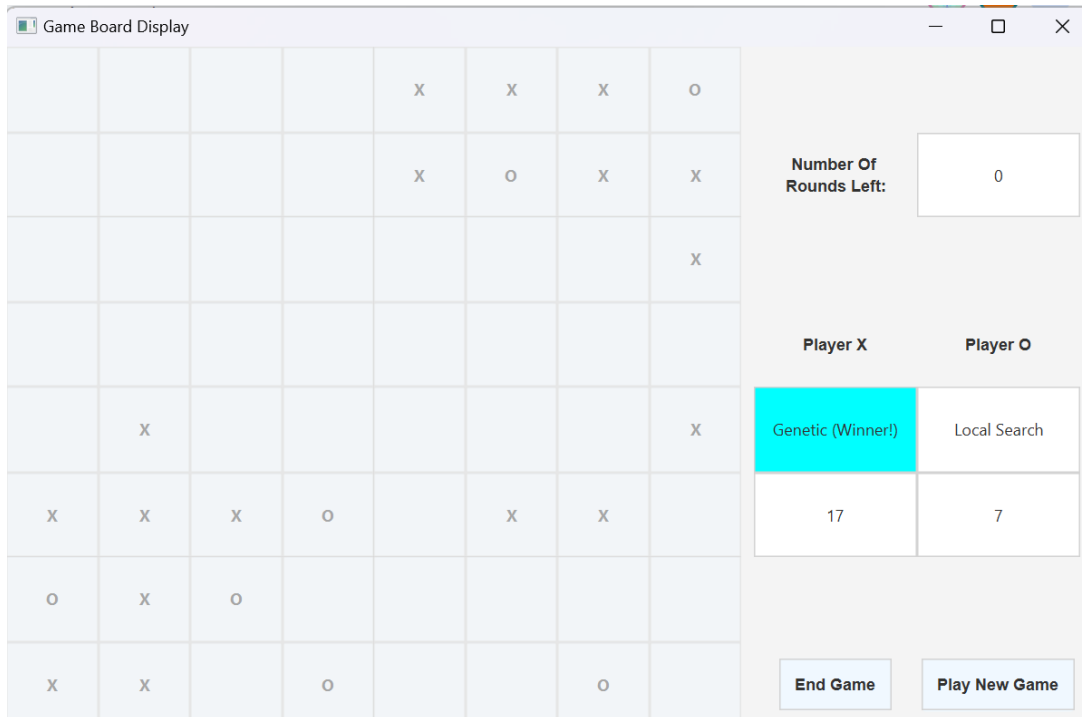
Didapatkan untuk permainan 28 ronde,

Game Board Display									
O	O	O	X	X	O	X	O	Number Of Rounds Left:	0
X	O	O	X	X	X	X	X		
O	X	X	X	X	X	O	X		
O	O	X	X	X	X	X	X	Player X	Player O
O	O	X	X	X	X	O	X	Genetic (Winner!)	Local Search
O	X	X	X	O	X	O	O	36	28
O	X	O	X	X	O	O	O		
X	O	O	O	X	X	O	O	End Game	Play New Game

Didapatkan untuk permainan 9 ronde,

Game Board Display									
X	O	X	O	X	O	X	O	Number Of Rounds Left:	0
X	X	X	X	X	X	X	O		
					X		O		
					X			Player X	Player O
	O							Genetic (Winner!)	Local Search
		O						17	9
X	X		O						
X	X							End Game	Play New Game

Didapatkan untuk permainan 8 ronde,



Kemenangan bot local search 0% dan kemenangan bot genetic minimax 100%.

Berdasarkan pertandingan diatas, dapat dilihat bahwa bot local search memiliki persentase kemenangan lebih tinggi dibandingkan bot genetic minimax. Hal tersebut dikarenakan bot genetic melakukan persilangan dan mutasi yang belum tentu menghasilkan hasil yang terbaik. Sedangkan local search walaupun sudah melakukan random, tetapi tetap mencari solusi yang terbaik.

Pengujian Bot Genetic vs Bot Minimax

Pengujian dilakukan untuk 3 permainan dengan jumlah ronde 28, 9, dan 8 ronde.

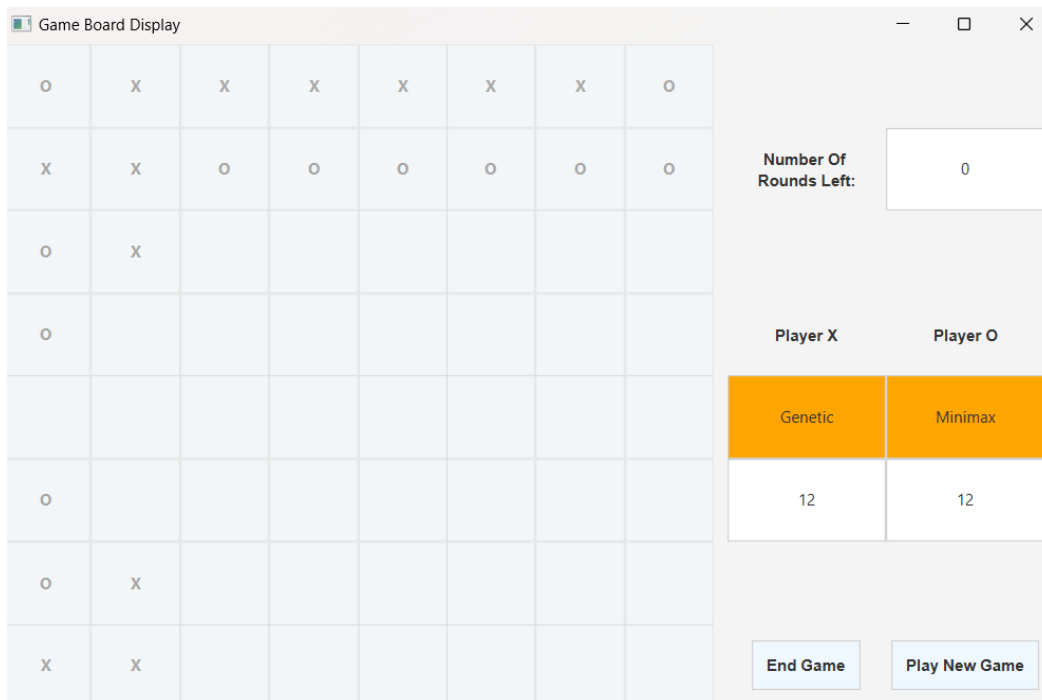
Didapatkan untuk permainan 28 ronde,

Game Board Display									
O	X	X	X	X	X	X	O	Number Of Rounds Left:	0
X	X	O	X	X	O	X	O		
O	X	O	X	O	X	O	X		
O	X	O	X	O	X	X	O	Player X	Player O
X	O	X	X	O	O	O	X	Genetic	Minimax (Winner!)
O	O	O	X	O	O	O	X	30	34
O	O	O	X	O	O	X	O		
X	O	X	O	O	X	O	O	End Game	Play New Game

Didapatkan untuk permainan 9 ronde,

Game Board Display									
O	X	X	X	X	X	X	O	Number Of Rounds Left:	0
X	X	O	O	O	O	O	O		
O	O	O							
O								Player X	Player O
		X						Genetic	Minimax (Winner!)
O								12	14
O	X								
X	X							End Game	Play New Game

Didapatkan untuk permainan 8 ronde,



Kemenangan bot minimax 100% dan kemenangan bot genetic 33.33%.

Berdasarkan pertandingan diatas, dapat terlihat bahwa bot minimax memiliki persentase kemenangan lebih besar daripada bot genetic. Hal tersebut disebabkan oleh minimax dapat melihat dua aksi kedepan sebelum bertindak, sedangkan genetic dilakukan persilangan dan mutasi yang belum tentu menghasilkan hasil yang maksimal walaupun sudah dimodifikasi dengan minimax.

Daftar Pustaka

K. K. Informatika. (2023). *Modul 3: Beyond Classical Search*. Sekolah Teknik Elektro dan Informatika | Institut Teknologi Bandung.

S. Russel and P. Norvig. (2010). *Artificial Intelligence A Modern Approach, 3rd ed.* Pearson.

Sebastian Lague. (2018). “Algorithms Explained – minimax and alpha-beta pruning,” Youtube. [Online]. Available: <https://youtu.be/l-hh51ncgDI?si=CVze3jab5EkDyhhI>

Genetic Algorithm. (2018, December 8). School of Computer Science | BINUS University.

Retrieved October 18, 2023, from <https://socs.binus.ac.id/2018/12/08/genetic-algorithm/>

Jain, S. (2023, April 20). *Introduction to Hill Climbing | Artificial Intelligence*. GeeksforGeeks.

Retrieved October 18, 2023, from

<https://www.geeksforgeeks.org/introduction-hill-climbing-artificial-intelligence/>

Lampiran

Repository GitHub: https://github.com/MHEN2606/Tubes1_AI_13521007

Kontribusi:

NIM	Nama	Pekerjaan
13521007	Matthew Mahendra	Algoritma Local Search, Minimax, dan Genetic Algorithm
13521010	Muhamad Salman Hakim Alfarisi	Algoritma Local Search, Minimax, dan Genetic Algorithm
13521013	Eunice Sarah Siregar	Algoritma Genetic Algorithm dan Local Search
13521018	Syarifa Dwi Purnamasari	Algoritma Genetic Algorithm dan Local Search