

Informe Proyecto II CI-5437

1. Introducción.

En el presente informe se presentan los resultados obtenidos luego de ejecutar los algoritmos negamax, negamax con poda alpha-beta, scout y negascout durante 15 minutos. Los algoritmos se ejecutaron siguiendo la variación principal de Othello 6x6. Las especificaciones de las computadoras en donde se ejecutaron los algoritmos fueron las siguientes:

- Computadora 1: i3 6ta generación, 12Gb de RAM y SO Ubuntu 23.04
- Computadora 2: i3 4ta generación, 4Gb de RAM y SO Mint 21

2. Resultados.

De la ejecución de cada algoritmo se registró la profundidad en la variación principal del problema, el valor devuelto por el algoritmo, cantidad de nodos expandidos y generados, el tiempo de ejecución de cada y la cantidad de nodos generados por segundo. Los resultados se presentan en las siguientes tablas.

negamax					
Profundidad	Valor	Expandidos	Generados	Tiempo (segundos)	Nodos/Segundos
34	-4	1	0	6,00E-06	0
33	-4	1	1	1,40E-05	71429
32	-4	2	4	1,60E-05	250002
31	-4	2	5	1,50E-05	333336
30	-4	4	12	3,50E-05	342857
29	-4	4	13	2,70E-05	481481
28	-4	27	90	0,000139	647482
27	-4	52	176	0,00254	69291
26	-4	305	1048	0,002043	512971
25	-4	1330	4497	0,007664	586769
24	-4	3381	11977	0,015907	752939
23	-4	21699	76825	0,052122	1473946
22	-4	119923	428401	0,244895	1749325
21	-4	953486	3478734	2,02528	1717656
20	-4	3619363	13078932	9,58393	1364673
19	-4	25526376	90647894	58,688	1544573
18	-4	251184784	876269597	578,815	1513903

negamax con poda alpha beta					
Profundidad	Valor	Expandidos	Generados	Tiempo (Segundos)	Nodos/Segundos
34	-4	1	0	9,00E-06	0
33	-4	1	1	1,00E-05	99999
32	-4	2	4	1,30E-05	307692
31	-4	2	5	1,30E-05	384615
30	-4	4	12	2,70E-05	444444
29	-4	4	13	2,30E-05	565242
28	-4	6	26	4,60E-05	565217
27	-4	20	81	0,00012	675000
26	-4	52	237	0,000371	638814
25	-4	234	1002	0,003227	310505
24	-4	350	1501	0,00422	355687
23	-4	900	4067	0,00787	516773
22	-4	2099	9129	0,010921	835912
21	-4	22734	98754	0,071843	1374581
20	-4	29515	127643	0,093451	1365882
19	-4	62587	267603	0,192833	1387745
18	-4	199087	1259429	0,956444	1316783
17	-4	482139	2031923	1,52429	1333029
16	-4	7176690	29501797	22,3765	1318428
15	-4	10625624	43574642	33,6297	1295719
14	-4	25626713	107642870	81,4396	1321751
13	-4	100835794	415909955	309,372	1344368

scout					
Profundidad	Valor	Expandidos	Generados	Tiempo (Segundos)	Nodos/Segundos
34	4	1	0	0,00E+00	
33	-4	1	0	0,00E+00	
32	4	2	0	0,00E+00	
31	-4	2	1	0,00E+00	
30	4	4	1	0,00E+00	
29	-4	4	1	0,00E+00	

28	4	27	6	0,00E+00	
27	-4	52	14	0,00E+00	
26	4	305	114	0,002327	48990
25	-4	1330	467	0,010329	45213
24	4	3381	1502	0,023264	64563
23	-4	21699	10138	0,140127	72349
22	4	119923	50977	0,789987	64529
21	-4	953486	463178	7,10761	65166
20	4	3619363	1756971	28,7013	61216
19	-4	25556376	11732255	161,323	72725

negascout					
Profundidad	Valor	Expandidos	Generados	Tiempo (Segundos)	Nodos/Segundos
34	-4	1	0	5,00E-06	0
33	-4	1	1	9,00E-06	111110
32	-4	2	5	1,20E-05	416663
31	-4	2	5	1,20E-05	416670
30	-4	6	18	2,90E-05	620690
29	-4	6	19	2,80E-05	678571
28	-4	8	32	4,60E-05	695654
27	-4	20	90	0,000107	841121
26	-4	86	418	0,000523	799235
25	-4	393	1837	0,003974	462255
24	-4	571	2703	0,004177	647115
23	-4	847	4243	0,00709	598449
22	-4	4751	13253	0,016526	801948
21	-4	10715	53622	0,037263	1439015
20	-4	18304	90324	0,064236	1406127
19	-4	41816	203353	0,138248	1470929
18	-4	140325	676994	0,466932	1449877
17	-4	264869	1267306	0,83461	1518441
16	-4	1705777	8226831	5,52802	1488206
15	-4	6128742	29511025	20,5767	1434196
14	-4	14453750	70789280	50,3575	1405735

13	-4	57293208	278665312	181,358	1536548
12	-4	146018878	711096072	476,711	1491671

3. Conclusiones.

Se concluye que el algoritmo con mejores resultados es el negascout ya que éste llega a mayor profundidad. Por otro lado, el algoritmo menos favorable fue scout, sin embargo, este genera menos nodos que negamax en todas las profundidades. Además, hubo profundidades en las que negamax con poda alpha beta género menos nodos que negascout.

En esté proyecto logramos comparar el comportamiento y eficiencia de estos algoritmos. Además se logró confirmar su resultado al ejecutarlos sobre la variación principal del juego. Se observó la diferencia de recorrer el árbol de juego con y sin poda, siendo los algoritmos que realizan poda los que tuvieron mejor desempeño.