

# Reporte: Algoritmo del Caballo (Ávido)

Oswaldo de Luna

27 de octubre de 2019

# Análisis

Probaremos diferentes casos para el tablero normal, el cual es de 8x8.

```
1      26      15      24      29      36      13      32
16     23      28      35      14      31      40      37
27     2       25      30      61      38      33      12
22     17      62      45      34      41      50      39
3      46      21      60      49      64      11      42
18     57      48      63      44      53      8       51
47     4       55      20      59      6       43      10
56     19      58      5       54      9       52      7

64
true
BUILD SUCCESSFUL (total time: 2 seconds)
```

Figura 1: Punto inicial en (0,0)

Para tal tablero con punto de inicio en (0,0) si existe un recorrido que solucione el problema.

```
27     30      17      34      45      64      15      36
18     33      28      57      16      35      44      61
29     26      31      46      63      60      37      14
32     19      58      1       56      47      62      43
25     2       55      52      59      42      13      38
20     51      22      41      48      53      10      7
3      24      49      54      5       8       39      12
50     21      4       23      40      11      6       9

64
true
BUILD SUCCESSFUL (total time: 2 seconds)
```

Figura 2: Punto inicial (3,3)

Para un punto céntrico el cual es (3,3), hay también una solución.

Ahora veamos qué sucede con tableros más grandes, pero con un mismo punto de inicio (0,0).

```

---      ---      ---      ---      ---      ---      ---      ---      ---      --
535      20      1375     554      1503     1332     1379     1382     1499     14
552      129     574      1329     1374     1377     1500     1331     1334     13
21       1326     553      1376     1195     1330     1333     1372     1281     11
130      575     1192     1325     1328     1373     1194     1197     1322     12
1189     22      1327     1156     1193     1196     1323     1280     1159     11
576      131     1188     1191     1324     1155     1158     1185     1112     11
23       1190     577      1154     1157     1186     1111     1152     1091     11
132      579     648      1187     640      1153     1090     1109     1034     65
649      24      641      578      651      1110     645      638      1089     10
580      133     650      647      644      639      652      1033     654      63
25       642     135      582      27       646      137      584      29       95
134      581     26       643      136      583      28       653      138      56

2500
true
BUILD SUCCESSFUL (total time: 2 seconds)

```

Figura 3: Tamano del tablero 50x50

242	805	668	823	812	867	902	981	988	97
667	44	811	808	815	878	813	972	905	98
736	243	804	669	824	809	866	901	970	97
45	670	737	810	799	814	879	826	865	90
244	735	672	803	742	825	798	863	880	82
671	46	741	738	797	800	743	828	795	86
702	245	734	673	802	739	796	769	862	82
47	674	703	740	731	744	801	776	773	76
246	701	676	733	708	775	730	745	770	77
675	48	707	704	729	732	709	774	727	74
682	247	700	677	710	705	728	697	718	77
49	678	683	706	699	690	717	712	725	69
248	681	250	689	686	711	698	691	716	71
251	50	679	684	253	52	687	714	255	54
680	249	252	51	688	685	254	53	692	71

10000  
true  
BUILD SUCCESSFUL (total time: 4 seconds)

Figura 4: Tamano del tablero 100x100

Podemos observar que sigue habiendo una solución, pero veamos qué pasa si tomamos como punto de inicio un punto que no esté tan al centro ni tan al borde de tablero, en este caso (30,71).

245	46	495	3824	3827	3844	1089	1186	38
496	447	3826	461	554	1185	3822	3807	10
47	246	497	560	3823	736	555	1090	11
448	559	460	553	498	561	1184	735	74
247	48	499	558	563	550	737	556	94
500	449	552	459	546	557	562	609	72
49	248	501	534	551	564	549	738	73
450	533	458	547	502	545	580	607	61
249	50	503	530	535	548	565	544	57
504	451	532	457	518	529	542	581	60
51	250	509	512	531	536	519	566	54
452	505	456	517	510	513	528	541	58
251	52	511	508	455	516	537	520	52
506	453	54	253	514	521	56	255	53
53	252	507	454	55	254	515	522	57

9664  
false  
BUILD SUCCESSFUL (total time: 3 seconds)

Figura 5: Tamano del tablero 100x100, con inicio en (30,71)

```

-----
3271  3268  3027  3260  3273  3266  3031  3022  3257  3034  30
90    3025  3272  3267  3030  3023  3258  3033  3018  2969  29
2983  3028  2977  3024  3259  3032  3021  2970  2987  2938  29
2978  89    2984  3029  2980  2971  2986  3019  2968  2943  29
2463  2982  2979  2976  2985  3020  2951  2972  2937  2954  29
88    2975  2464  2981  2950  2973  2934  2953  2944  2929  28
2465  2462  2949  2974  2933  2952  2945  2930  2829  2936  29
2948  87    2594  2823  2946  2931  2828  2935  2820  2599  28
2461  2466  2947  2932  2827  2824  2821  2598  2449  2818  24
86    2595  2458  2593  2822  2597  2826  2819  2472  2439  25
2457  2460  2467  2596  2825  2592  2471  2448  2589  2450  24
2468  85    2456  2459  2470  2447  2590  2451  2438  2441  20
2455  82    2469  2446  2591  2452  2171  2442  2085  2174  21
84    2445  80    2453  2170  2443  78    2173  2168  2087  76
81    2454  83    2444  79    2172  2169  2086  77    2084  20

10000
true
BUILD SUCCESSFUL (total time: 3 seconds)

```

Figura 8: Tamano del tablero 100x100, con inicio en (36,99)

Los dos casos anteriores toman como casilla inicial las que estén apegadas al borde del tablero.

En los siguiente casos, tenemos los valores más elevado, por lo que imprimir la matriz es muy tardado y se evitó hacerlo.

```

run-single:
801328
false
BUILD SUCCESSFUL (total time: 3 seconds)
|

```

Figura 9: Tamano del tablero 1000x1000, con inicio en (0,0)

```

run-single:
102684895
false
BUILD SUCCESSFUL (total time: 1 minute 57 seconds)
|

```

Figura 10: Tamano del tablero 10000x10000, con inicio en (0,0)

Podemos observar que el algoritmo, como fue programado, procura tomar caminos que cruzan por el borde del tablero, ya que estas casillas son las que menos posibilidades tienen de movimientos una vez que cae el caballo. Por lo que, termina yendo al centro, porque estas casillas son las que más posibilidades tienen en un principio.

Esta razón es la que explica por qué al elegir como casilla inicial alguna casilla céntrica, el algoritmo no puede hallar una solución para el problema. Mientras que si le damos como inicio una casilla al borde, el algoritmo lo solucionará sin mayor esfuerzo.

Sin embargo, al aumentar de manera exponencial el número de casillas, su comportamiento se vuelve más errático y ocasiona que el algoritmo no logre resolver los casos.

Caso especial, donde podemos observar que no puede tocar el centro, pero este tipo de movimiento permite tocar todos los cuadros alrededor de uno.

```
- - -  
run-single:  
1      6      3  
4      0      8  
7      2      5  
  
8  
false  
BUILD SUCCESSFUL (total time: 2 seconds)  
|
```

Figura 11: Tamano del tablero 3x3, con inicio en (0,0)