AI Intro Homework #1

NCTU Spring 2018

0516076 施威綸

1. **performance measures**

    (a) speed, fuel consumption

    (b) false non-filtered rate, false filtered rate, filter speed

    (c) click rate, correlation (with target)

2. **graph search BFS (L->B)**

    Frontier:{L}, Explored:{}

    -> Frontier:{M, T}, Explored:{L}

    -> Frontier:{T, D}, Explored:{L, M}

    -> Frontier:{D, A}, Explored:{L, M, T}

    -> Frontier:{A, C}, Explored:{L, M, T, C}

    -> Frontier:{C, S, Z}, Explored:{L, M, T, C, A}

    -> Frontier:{S, Z, P, R}, Explored:{L, M, T, C, A, C}

    -> Frontier:{Z, P, R, F}, Explored:{L, M, T, C, A, C, F}

    -> Frontier:{P, R, F, O}, Explored:{L, M, T, C, A, C, F, Z}

    -> Frontier:{R, F, O, B}, Explored:{L, M, T, C, A, C, F, Z, P}

    solution: L -> M -> D -> C -> P -> B

3. A* search, f(n) = g(n)+h(n)

      Frontier:{L(244=0+244)}, Explored:{}

  -> Frontier:{M(311=70+241), T(450=111+329)}, Explored:{L}

  -> Frontier:{D(387=145+242), T(450=111+329)}, Explored:{L, M}

  -> Frontier:{C(425=265+160), T(450=111+329)}, Explored:{L, M, D}

  -> Frontier:{T(450=111+329), P(501+403+98), R(604=411+193)},

    Explored:{L, M, D, C}

  -> Frontier:{P(501+403+98), A(595=229+366), R(604=411+193)},

    Explored:{L, M, D, C, T}

  -> Frontier:{B(504=504+0), A(595=229+366), R(604=411+193)},

    Explored:{L, M, D, C, T, P}

  -> Frontier:{A(595=229+366), R(604=411+193)},

    Explored:{L, M, D, C, T, P, B}

    solution: L -> M -> D -> C -> P -> B

4. chessboard

  (a) b = 7+7 = 14

  (b) shortest path: draw a rectangle with point A & B on the diagonal vertices

  Heuristic = the # of axis that does not match B(x, y) + 1 if ->

  -> if the minimum path to B(x, y) is blocked by x on the next step

the constraint x block is removed

the minimum cost between A and B is # of axis that match B(x, y)

but if one of the axis matches with **x** block between the square and b,

one or more move is needed

(addition cost) >= (1), if the shortest path is blocked

So,

(minimum cost + addition cost) >= (my heuristic)

=> admissible

(the cost between any 2 squre C, D) >= (the shortest path between C, D)

h(C) >= c(D, a, C) + h(D)

=> consistent

(c) step 0: 1

step 1: 8

step:2: 29
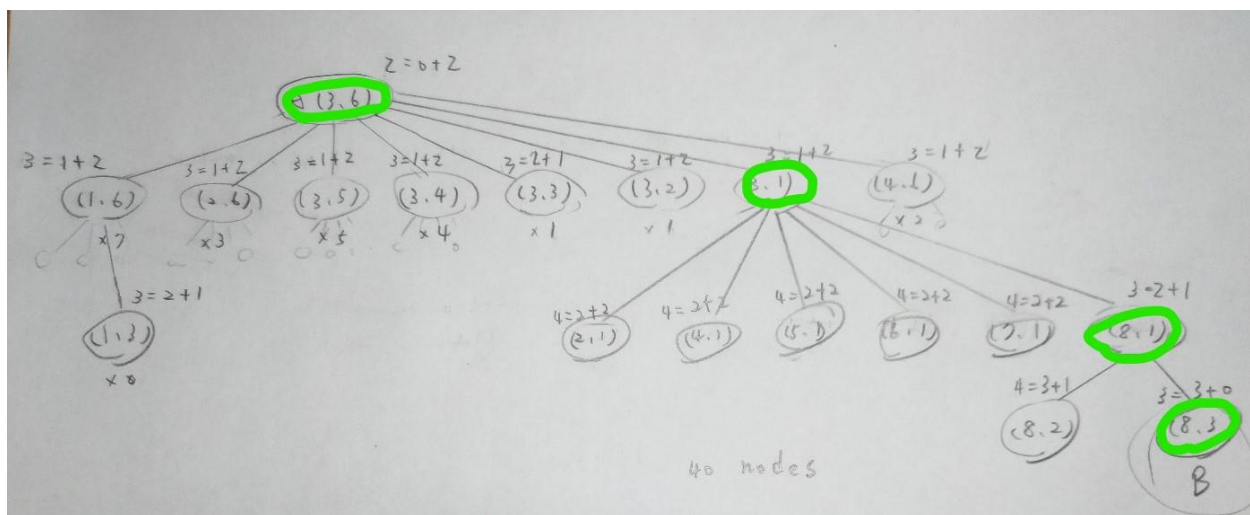
-> 1+8+29 = 38 nodes

solution depth = 3

(d) step 0: 1

step 1: 9

step 2: 38

-> 1+9+38 = 48 nodes

solution depth = 3

(e) A*: f(n) = h(n)+g(n)

-> 4 nodes