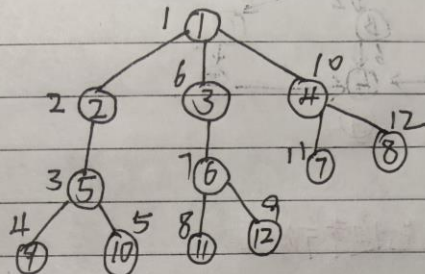


21200100001 王志强

第二章 章节作业

2.1

(1) 根据深度优先遍历列出图中树的节点访问序列 (在所有情况中都选择最左支优先访问)



(2) 根据深度优先搜索策略搜索时从初始状态迭代至搜索结束的过程中 OPEN 表和 CLOSED 表中所存储的内容

OPEN 表

CLOSED 表

{1}

{}

{2, 3, 4}

{1}

{5, 3, 4}

{1, 2}

{9, 10, 3, 4}

{1, 2, 5}

{10, 3, 4}

{1, 2, 5, 9}

{3, 4}

{1, 2, 5, 9, 10}

{6, 4}

{1, 2, 5, 9, 10, 3}

{11, 12, 4}

{1, 2, 5, 9, 10, 3, 6}

{12, 4}

{1, 2, 5, 9, 10, 3, 6, 11}

{4}

{1, 2, 5, 9, 10, 3, 6, 11, 12}

{7, 8}

{1, 2, 5, 9, 10, 3, 6, 11, 12, 4}

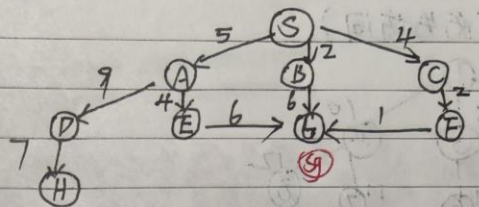
{8}

{1, 2, 5, 7, 10, 3, 6, 11, 12, 4, 7}

{}

{1, 2, 5, 9, 10, 3, 6, 11, 12, 4, 7, 8}

2.2 请根据下图写出利用代价树的宽度优先搜索(BFS)和深度优先搜索(DFS)找到从初始节点S到达目标节点G的路径,并分别列出两种情况下 OPEN 表和 CLOSED 表的变化情况。



(1) 宽度优先搜索示例表

OPEN 表

CLOSED 表

{S:0}	{}
{A:5, B:2, C:4}	{S:0}
{B:2, C:4, D:(5+9)}	{S:0, A:5}
{C:4, D:(5+9), G:(2+6)}	{S:0, A:5, B:2}

G 结点已经出现, 路径为 S→B→G

(2) 深度优先搜索

OPEN 表

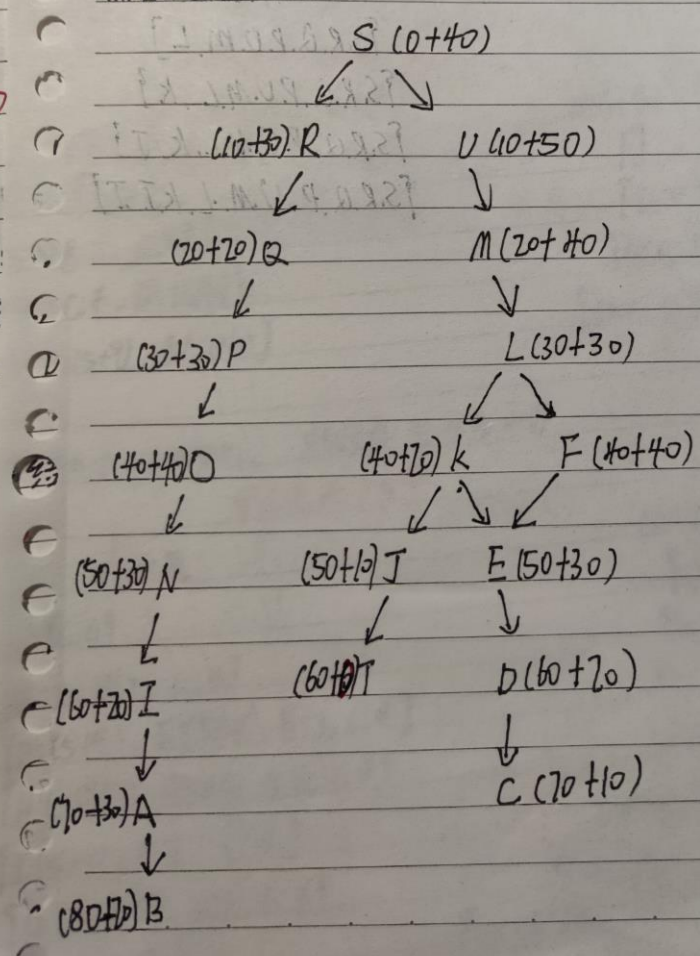
CLOSED 表

{S:0}	{}
{A:5, B:2, C:4}	{S:0}
{D:(5+9), E:(5+4), B:2, C:4}	{S:0, A:5}
{H:(5+9+7), E:(5+4), B:2, C:4}	{S:0, A:5, D:(5+9)}
{E:(5+4), B:2, C:4}	{S:0, A:5, D:(5+9), H:(5+9+7)}
{G:(5+4+6), B:2, C:4}	{S:0, A:5, D:(5+9), H:(5+9+7), E:(5+4)}
{B:2, C:4}	{S:0, A:5, D:(5+9), H:(5+9+7), E:(5+4), G:(5+4+6)}

G 结点出现 路径为 S→A→E→G

2.3 A\* 算法寻路: 机器人从图(a)中的S出发, 利用A\*寻找一条到达T点的最短路径, 每次只能向与当前位置相邻的上下左右四个邻域块移动(一次的代价为10), 墙壁处不可移动。用G表示S到当前点的路径长度, H表示当前点到T的曼哈顿距离(两点间的水平距离和垂直距离之和),  $F = G + H$ 。如果F相同, 优先考察H小的。

(1) 用A\*算法搜索, 在图(b)中画出搜索树, 并标出扩展节点的G+H。





Date / /

OPEN 表

closed

{S}

{}

{R, U}

{S}

{Q, U}

{S, R}

{P, U}

{S, R, Q}

{U, D}

{S, R, Q, P}

{M, D}

{S, R, Q, P, U}

{L, D}

{S, R, Q, P, U, M}

{K, F, D}

{S, R, Q, P, U, M, L}

{J, E, F, D}

{S, R, Q, P, U, M, L, K}

{T, E, F, D}

{S, R, Q, P, U, M, L, K, J}

{E, F, D}

{S, R, Q, P, U, M, L, K, J, T}

(0+0+0) M

(0+0+0)

↓

(0+0+0) L

(0+0+0)

↓

(0+0+0) F

(0+0+0) K

(0+0+0)

↓

(0+0+0) E

(0+0+0) T

(0+0+0)

↓

(0+0+0) D

(0+0+0)

(0+0+0)

↓

(0+0+0) C

(0+0+0)

INTI

TO /

if