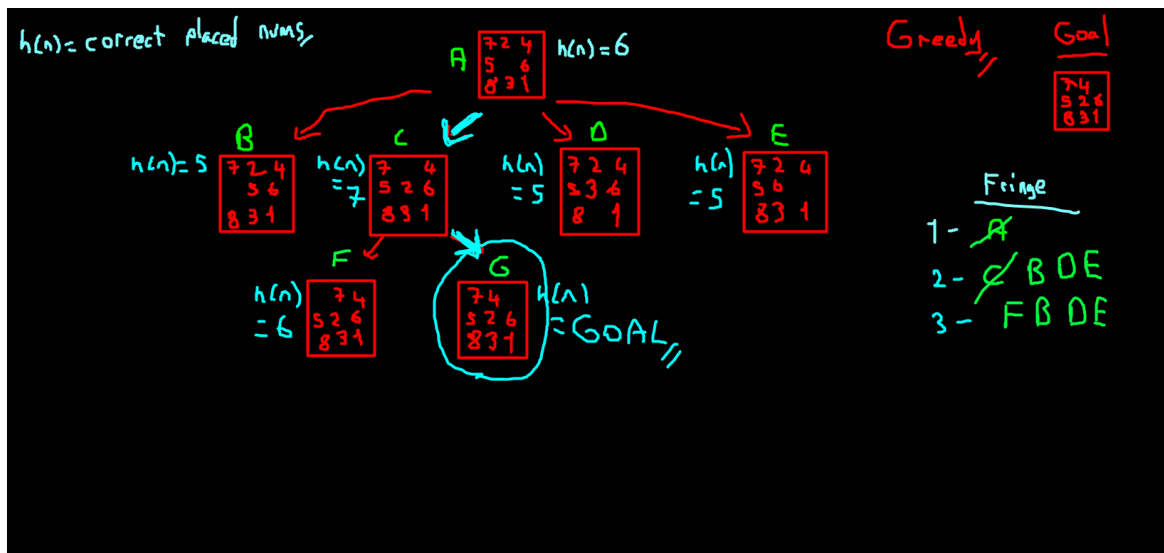# ARTIFICIAL INTELLIGENCE ASSIGNMENT -2

REPORT

Greedy vs A* Algorithm

-Greedy Best-First Algorithm choose the best option that gives the best profit fort he current step. It evaluates nodes by using just the heuristic function; that is, f(n) = h(n).
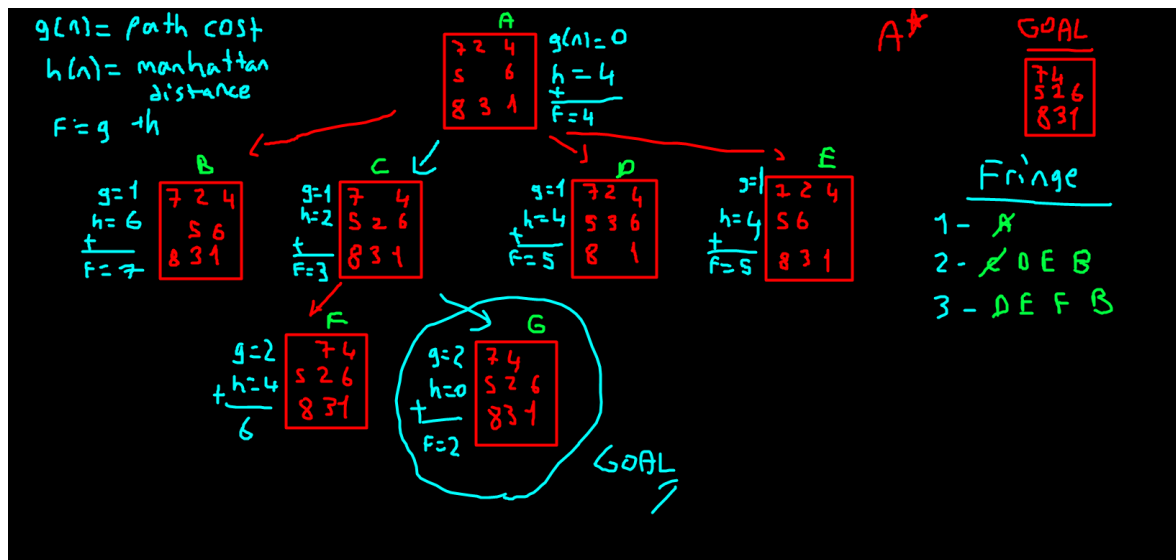
(I chose correct placed num for h(n))

+A* algorithm evaluates nodes by combining g(n), the cost to reach the node, and h(n), the cost to get from the node to the goal:

f(n) = g(n) + h(n)

(g(n) = state's level on the tree, h1(n) = manhattan distance and h2(n) = misplaced num)

- A* algorithm more powerful than greedy algorithm.

Sample initial and final state

```
int initialState[N][N] =
{
    {7, 2, 4},
    {5, 0, 6},
    {8, 3, 1}
};
```

```
int goalState[N][N] =
{
    {2, 5, 0},
    {8, 7, 4},
    {3, 1, 6}
};
```

GREEDY


```
2  0  5
8  7  4
3  1  6

2  5  0
8  7  4
3  1  6

Expanded nodes: 1019
```

Greedy outArt

A* with misplaced heuristic


```
Microsoft Visual Studio Debug Console
8  3  1

2  5  4
8  7  6
0  3  1

2  5  4
8  7  6
3  0  1

2  5  4
8  7  6
3  1  0

2  5  4
8  7  0      A* with misplaced num
3  1  6

2  5  0
8  7  4
3  1  6


Explored node: 32
```

A* with manhattan distance


```
Microsoft Visual Studio Debug Console
8  3  1

2  5  4
8  7  6
0  3  1

2  5  4
8  7  6
3  0  1

2  5  4
8  7  6
3  1  0

2  5  4
8  7  0
3  1  6
                A* with manhattan distance
2  5  0
8  7  4
3  1  6


Explored node: 19

C:\Users\hakan\source\repos\AI\Debug\AI.exe (process 2900) e
```

SUM

GREEDY < A*(misplaced) < A* (manhattan distance)