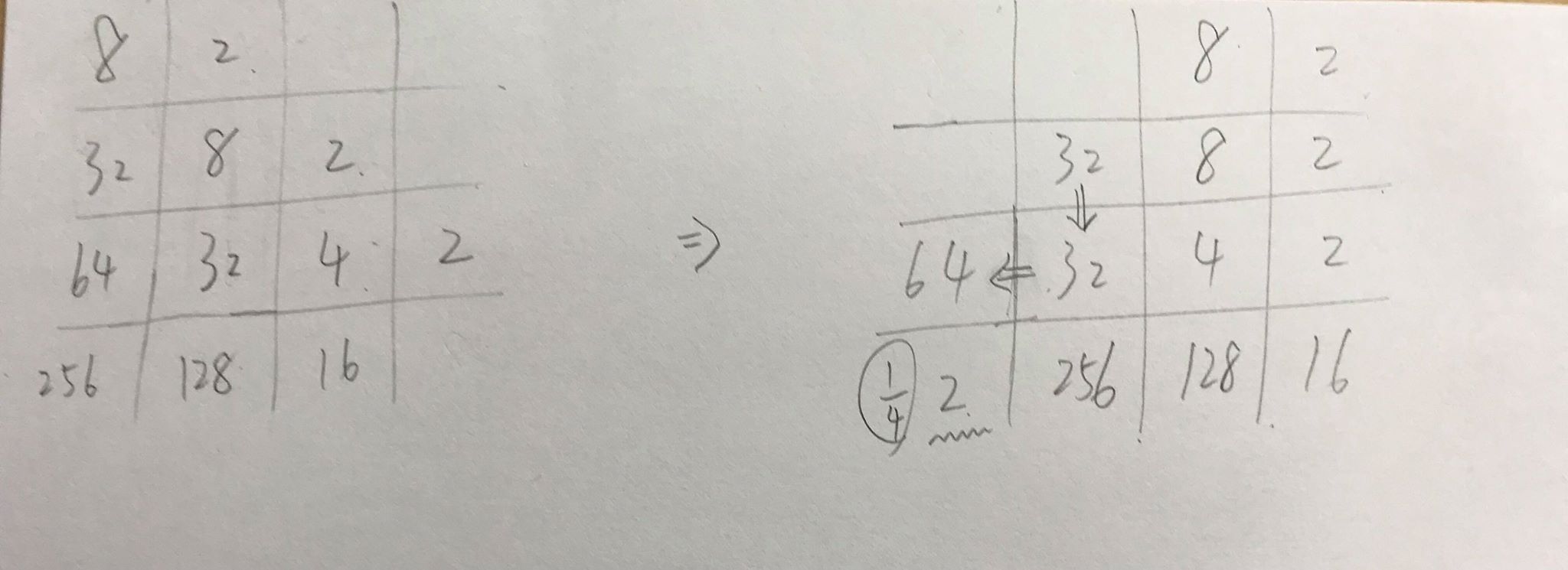
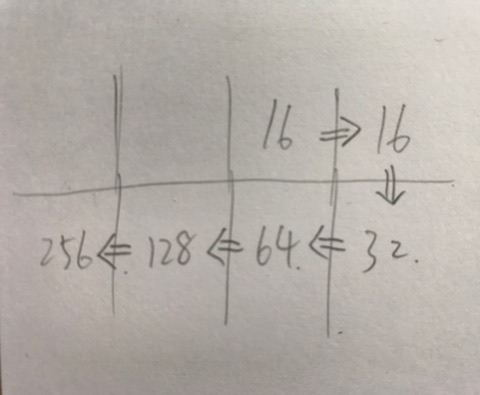
[2048 AI Proposal] 106703017邱奕修

首先先探討如何判斷棋盤的好/壞

1. 若隨機生成的數字(2,4)的四周都是大數字，則該數字會很難與其他小數字結合形成較大數字，以下圖為例: 雖然左圖往右、往下、往左滑結合出一個128，但仔細看可以發現第一步的往右時，有1/4的機率會產生新數字在左下角(假設產生新數字為2)，則未來這個2很難再與其他2接觸，且新生成的128也會因為這個2而與256隔開，也許未來因此無法結合。

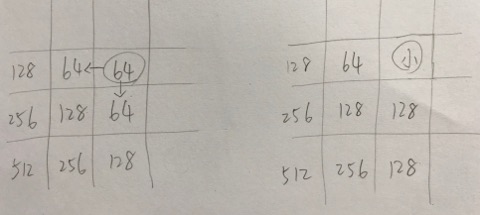


1. 若大的數字在中央，不管往哪個方向移動，都可能導致1.的發生。反之，若最大數字在四個角落，則至少有兩個方向可使最大數字不移動
2. 若能將大的數字排列成一排且照順序，則可用很少的空間(較少的步數，產生較少的數字)，將數字結合。階梯狀的排列也能達到此效果。

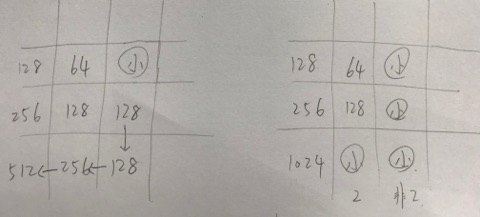


再來，討論怎麼移動是最佳解

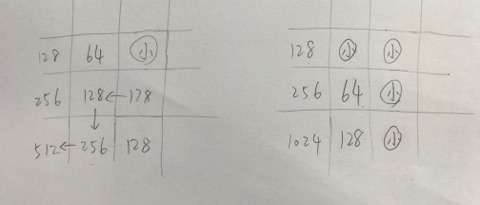
由上面分析，我認為階梯狀的排列會是最佳的狀況，因此就以階梯狀為例(下圖)



我們可以很輕易的合成出1024，且有很多種組合法，但怎樣才是最佳解呢?第一步我們有兩種走法(下、左)可將兩個64合成成128，且可以發現，不管走下還走左，都會使左上角補一格小數字進來。因此第一步兩種走法一樣(我們以往下繼續探討)



第二步也有兩種走法(下、左)可將兩個128合成出256，若選擇向下，可以看到之後的走法如右邊，且小數字會靠近最大數字(1024)若剛好中下的數字為2，它右邊的數字又非2，由上面分析可知，這是不好的情況。



若選擇走左，則能讓較大數字依然集中在最大數字旁

總結

我認為2048得最佳技巧為讓最大數字固定在一角落，且向周圍遞減(階梯狀)，原因為可避免掉產生2在一群大數字中間，有效運用這16格，才是2048高分的關鍵。實行方面，目前想法為:用遞迴方式討論未來兩、三步各走法，用計分方式找出最佳解，若移動數字n的四周最小數字為2kn，則為k分，如2的四周最小數字為8則為2分(22\*2)，最小分數者為最佳解。