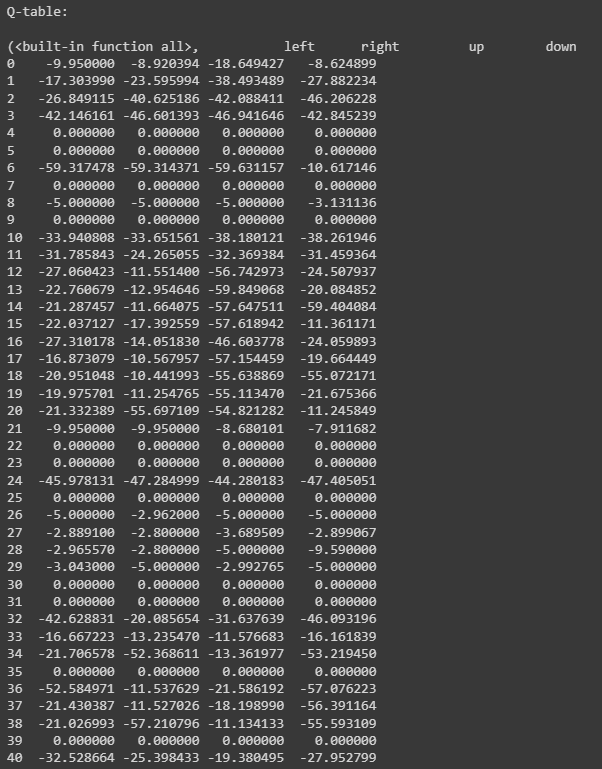
Maze

資管三 A 黃彥程 109403526

Colab連結:

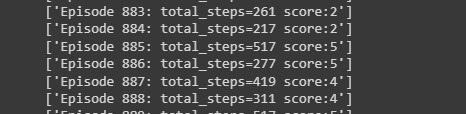
<https://colab.research.google.com/drive/1cKGbKfO5iy12-ELZV04tUcAFAtfWGxfI>

Q-table:

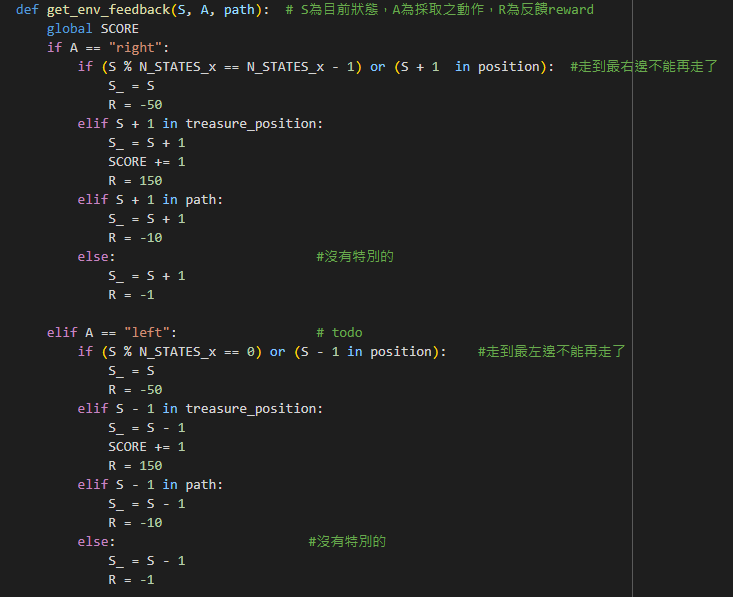


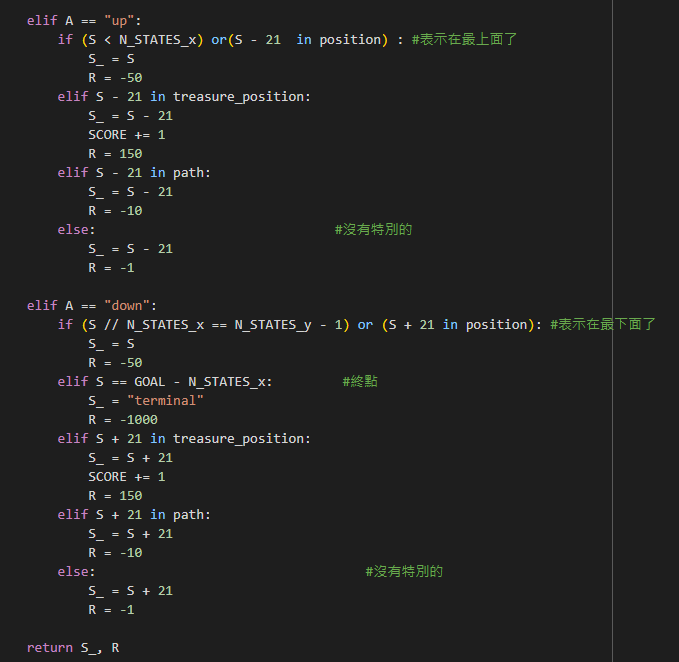
最佳:

score:5 step:277



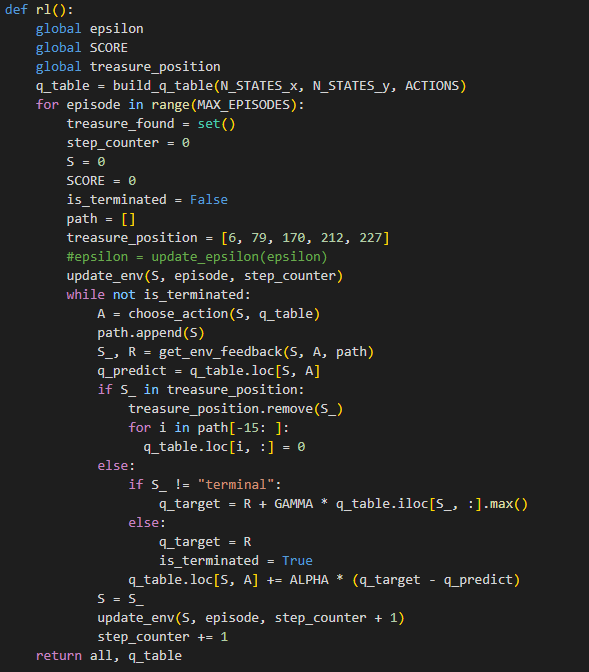
過程:





四個方向方法其實都差不多，就是用他那個方向向前一格後判斷是否是在牆壁上 or 出界(這兩個我寫在一起) or 拿到寶藏又或者是重複走之前走過的路，而成功出去我只有寫在”down”因為只有這種方式出的去，以減少判斷次數。

撞到牆壁或出界就不讓他走，然後扣它分數，這樣下次它就不太會再撞一次，重複走之前的路會浪費時間所以也扣它分，拿到寶藏就SCORE += 1，然後reward給高一點引誘它來拿，具體判斷它有沒有重複拿到我寫在rl():



(treasure\_found我忘記刪掉了，那是之前還在測試的寫法)

拿到一次就從裡面刪掉，然後一個episode結束就再重新賦予treasure\_position裡面的位置。

心得:

這份作業首先要先把地圖放進去，但是我後來是直接在reward用判斷的方式，不是真的放位置在地圖中，因為我一開始查資料多數是有看到人家有真的放位置在圖裡面，後來看了程式碼還是覺得這樣多此一舉，我直接判斷有沒有撞牆比較快，然後就是我一開始在設置有沒有走出迷宮時，設錯方向(因為一開始預設是向右有出去的方法，但是作業地圖是要向下)，然後我就一直不知道為甚麼沒有走出來，十分懊惱。

其實在做這份作業時，一開始當然是只有寫到reward部分利用Reward比例去鼓勵機器人去拿寶藏跟出去，但是後來發現這樣雖然跑得出來但是步數下降速度不快，然後去研究Epsilon Greedy 加上了遞減(把作業程式碼的 > EPSILON改成 <，然後加上遞減)讓機器越來越按照之前學習的方式走，這樣就比較會有學習成效，可是我發現雖然這樣有辦法把步數壓到不超過100，SCORE卻是一直歸零狀態，然後我就又設置了如果SCORE沒有5不給它出去的規定，想當然而直接變成幾萬步在跳。後來看到google上有人設置走回頭路就扣分，才又把path判斷補上，確實是快了不少，但是一樣SCORE都沒有甚麼辦法拿到5分，試了好幾次不同組合之後發現關鍵問題是，如果每次想拿寶藏走回頭路又都知道要扣大分，是我我也不拿，才又加上如果拿到寶藏最後15筆Q-table上的資訊重置，然後把我前面雞婆弄的遞減 EPSILON 改回助教寫的模樣，果然功夫不到家還是不要亂搞比較好，害我弄了好久。