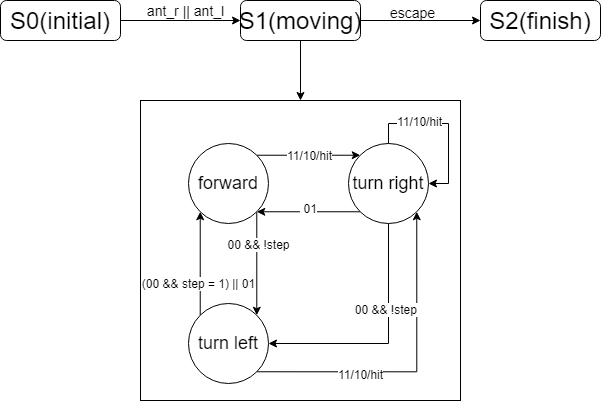
EECS1010 Design Assignment 03

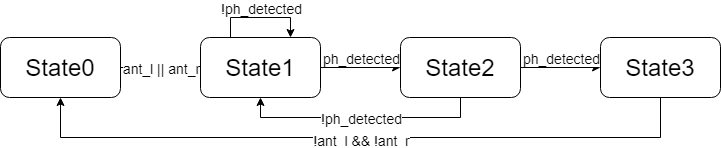
Student ID: 107070002 Name: 莊英暄

1. Design Concept and Description



我的想法是在一開始的狀態為initial，在這個state時會一直往前走直到觸角感覺到任何東西，也就是lr != 00，並開始進行moving階段。

在moving階段時，我盡量以靠左側行走，有三種移動方式，當前方遇到牆或是右側有牆時，會選擇向右轉直到牆壁在左側，而牆壁在左側時則向前走。當遇到轉角的時候會有兩個步驟，會先往前到空地，這時step = 0代表在第一步驟，會向左轉然後將step設為1，在step = 1時，會向前走並把step設回0，這樣就能在遇到轉角時順利做完轉向以及向前。

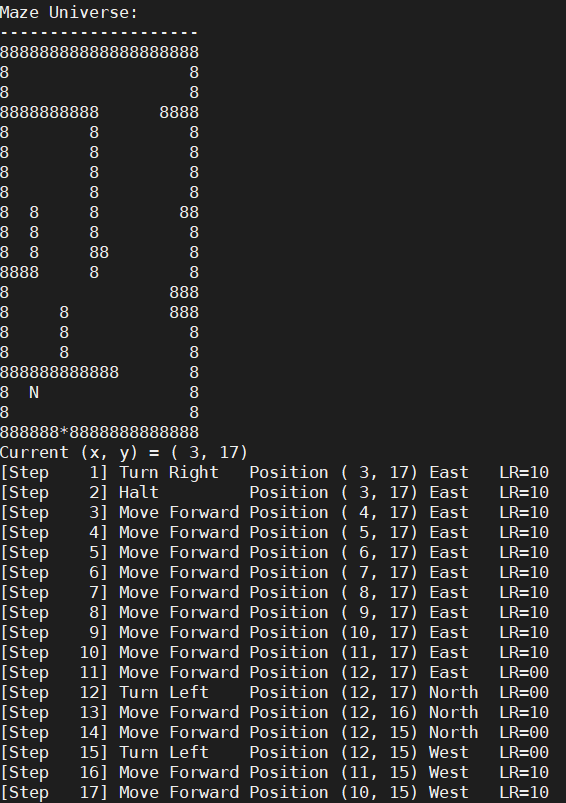
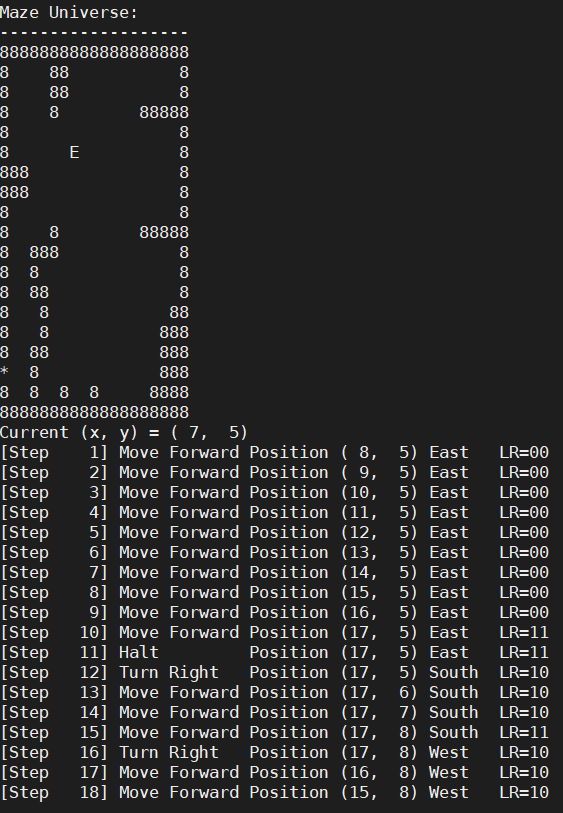
最後在到達終點時escape會被設為1，這時就代表順利逃跑，並結束程式。

Challenge2:

因為多了會偵測掉落物的東西，因此只用3個state不夠，需要新增一個state來讓偵測物品更加完整，如圖片所示，state0為initial state，只會往前，state1以及state2的行為模式與普通模式的行走一樣，state3則是背向牆面並回到state0，重新出發。

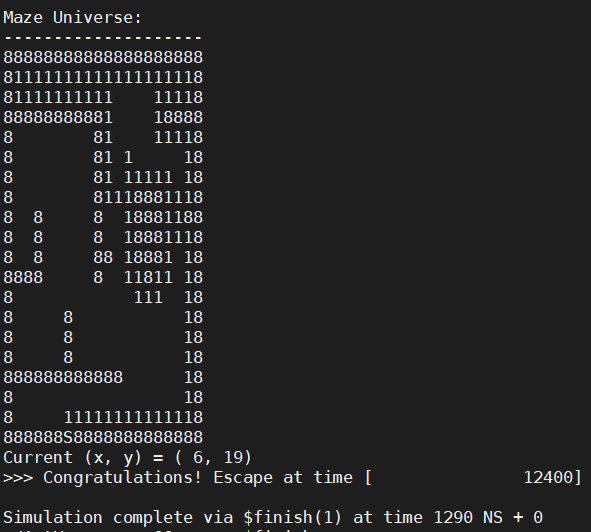
1. Simulation and Discussion

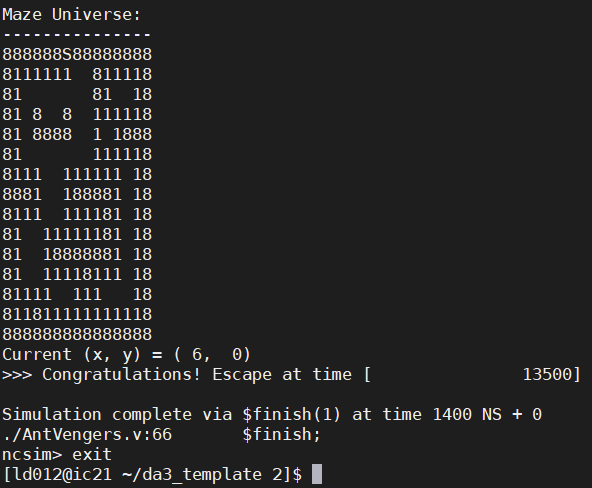
第一迷宮 第二迷宮



最麻煩的部分是經過轉角得時候，因為不能同時轉彎並前進，因此需要拆成兩個階段執行，也就是step0為轉彎而step1為直走。

Challenge discuss:





一開始先正常行走，當遇到重複走過的點的時候會選擇遠離牆壁並回到state0直走直到靠近牆壁。判斷方式為當現在動作為前進並且踏在地點為已經經過的地區時會向右轉直到背向牆壁，接著進入state0一直往前直到到達下一個牆壁附近。

我將繞牆拆成兩個部分，state1 代表一般的繞牆，當採到已經走過的點時會先進入state2進行判斷，state2時會判斷你在繼續下一步的時候是否還是在已經踩過的點，如果是的話就代表開始循環的然後進入state3，如果不是就代表可能只是在轉彎而已，所以回到state1。

1. Summary

這次作業花了我不少時間思考，也讓我更了解硬體code執行的邏輯，最一開始在寫的時候一直覺得很難，但是後來仔細思考後發現其實沒想像中的難，只是在遇到處理轉角時卡了非常的久。最後使用了reg step當作判斷條件來執行轉彎動作。

在處理challenge上，原本只打算處理possible solution，但是後來發現好想可以直接寫出來，雖然花了不少時間，但是最後還是打出來了，我覺得最麻煩的是要在甚麼情況下進入state3是最複雜的，因為轉向就會使地面留上物品，導致如果只用一個state行走會常常跑出wall island，最後千辛萬苦才終於寫出一個能跑也比較理想也穩定的走法。