

The background features a complex network of thin grey lines connecting various circular nodes. The nodes vary in size and color, including dark blue, light blue, and grey. Some nodes are highlighted with larger, concentric circles. A large black rectangle is positioned in the lower right, containing the title and author information.

# SCRATCH比賽AI組快速攻略

---

和順國中 林信廷

# 講師簡介

- 臺南市和順國中資訊組長
- 臺南市資訊組長分區召集人
- 臺南市科技輔導團兼任輔導員
- 和順科技中心課程設計教師
- 專長：機電整合、人工智慧

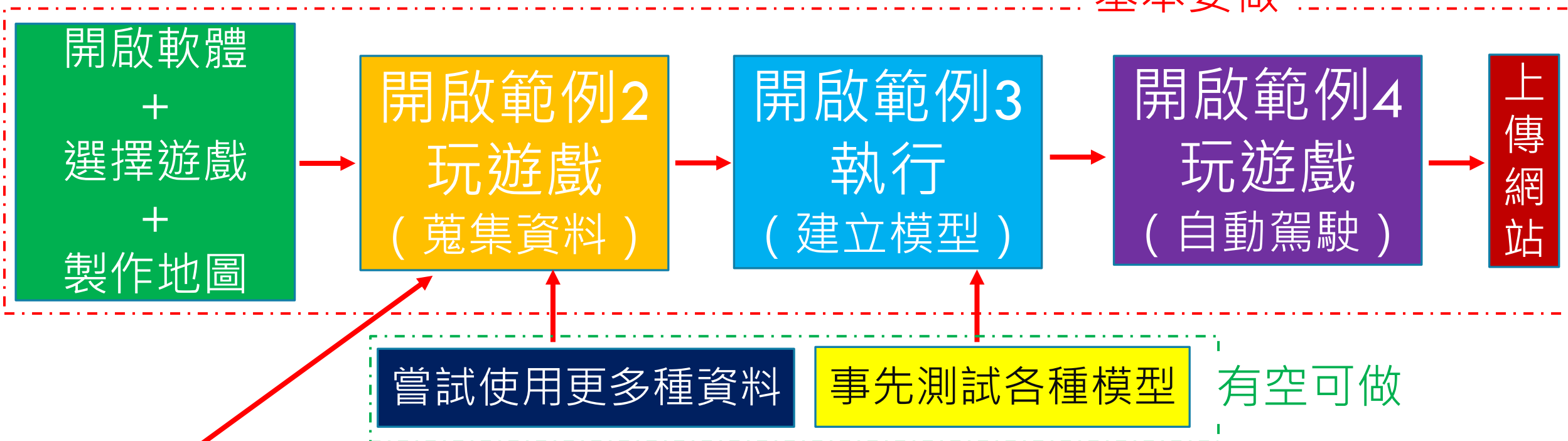




# SCRATCH競賽AI組-快速攻略

## ■先講結論：比賽流程整理

基本要做



參考範例1  
製作自動蒐集資料的程式

學生很OK再做

The background of the slide is a complex, abstract network diagram. It consists of numerous nodes of varying sizes and colors (dark blue, light blue, and grey) connected by thin, light grey lines. Some nodes are highlighted with larger, concentric circles. The overall aesthetic is modern and technological.

# 競賽規則整理

---



# 競賽規則整理

## ■ 相關日程（國中組）：

■ 報名期限：9/27

■ 預賽-AI自選賽：10/14

AI迷宮賽：10/15（必選、全國）

■ 決賽-AI自選賽：11/16

AI迷宮賽：11/17（必選、全國）



# 競賽規則整理

## ■ 相關日程（ 國小組 ）：

■ 報名期限：9/27

■ 預賽-AI自選賽：10/13

AI迷宮賽：10/15（ 必選、全國 ）

■ 決賽-AI自選賽：11/15

AI迷宮賽：11/17（ 必選、全國 ）



# 競賽規則整理

## ■ 相關日程：

預賽：

1. 國小迷宮：110年10月15日(五)下午13:00-16:10
2. 國中迷宮：110年10月15日(五)下午13:00-16:10
3. 國小自選：110年10月13日(三)下午13:00-16:10
4. 國中自選：110年10月14日(四)下午13:00-16:10

決賽：

1. 國小迷宮：110年11月17日(三)早上9:00-12:10
2. 國中迷宮：110年11月17日(三)下午13:0-16:10
3. 國小自選：110年11月15日(一)下午13:00-16:10
4. 國中自選：110年11月16日(二)下午13:00-16:10



# 競賽規則整理

- 兩人一隊
- 競賽平台：PAIA網站的「AI競技場」
  - <https://docs.paia-arena.com/zh-tw>
- 比賽時間3hr





# 競賽規則整理

- 要上傳「AI程式」與「模型資料」兩個檔案
  - 因為每個人選擇用來訓練的項目不同，所以要有對應的AI程式才看的懂自己的模型
- 檔名是有限制的：
  - AI程式的名稱：`ml_play.py`（python檔）（遊戲架構的限制）
  - 模型資料的名稱則沒有限制，英文即可：`model.pickle`（python特有資料格式）
  - 但要注意AI程式呼叫模型檔，檔名要一致



# 競賽規則整理

- 採排名制，過的關卡越多、過關時間越短，排名越前面。
- 每輪淘汰後50%的隊伍。



# 競賽規則整理

## ■ 預賽：

- 比賽開始時，隊伍各自上網，**依序**挑戰每個關卡（不能跳關）（到時候也沒有「小試身手」跟「經典迷宮」可以選）
- 關卡**可重複挑戰**，取最佳成績

## ■ 決賽：

- 製作時間結束後，在AI平台競賽**統一挑戰**，並開放觀戰



# 競賽規則整理

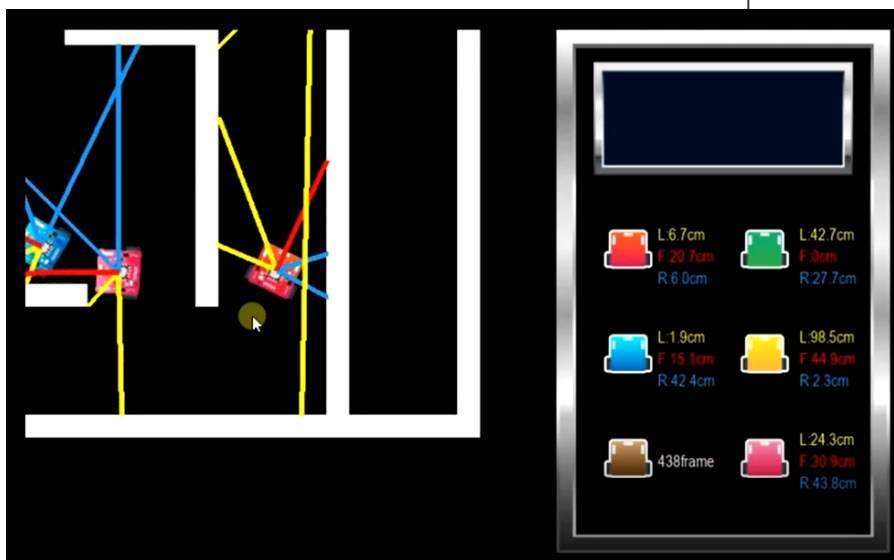
## ■ 相關參數：

- 預賽：因為是自行挑戰，還是會有調fps、感測器數量等參數的選項
- 決賽：由管理者統一開啟遊戲，沒有調參數的選項。



# 競賽規則整理

## ■ 遊戲畫面



### 可選擇程式

編號	A I 的名字	說明	狀態	使用積木	上傳新增
1	Untitled		待命中 <span>出賽</span>	  	
2	Alcar0912	Alcar0912	待命中 <span>出賽</span>	  	
	Al-car1		待命中 <span>出賽</span>	  	
	Alcar1	Al-car1	待命中 <span>出賽</span>	  	



# 競賽規則整理

- 比賽用的地圖並非內建的那些：
- 預賽：
  - 比賽開始時，上網挑戰每個關卡即可看到地圖
- 決賽：
  - 製作時間結束後，在AI平台競賽才會看到地圖



# 競賽規則整理

## ■地圖為：

- 預賽：30格 \* 30格（含邊界）

- 決賽：40格 \* 40格（含邊界）



# 競賽規則整理

- 可以使用不同的AI來挑戰不同的關卡
  - 預賽時，看的到地圖，所以可以針對地圖訓練不同的AI
  - 但決賽時，這個方法或許就不好用了，因為學生看不到地圖，所以還是要訓練出能應付所有狀況的AI





# 競賽規則整理

## ■ 比賽地點：

- 預賽：在各自學校進行

- 決賽：到資訊中心統一進行（但會視疫情調整）



# 人工智慧簡介

---



# 人工智慧簡介

- 什麼叫做智慧？
  - 會思考、會學習、會記憶？
- 什麼叫做會思考？
- 什麼叫做人工智慧？
  - 電腦根據我們提供的資料集，做出判斷



# 人工智慧簡介

- 電腦根據我們提供的資料集，做出判斷
  - 例如我給電腦很多「距離感測器」跟「馬達轉速」的數據。
  - 他學起來之後，就會知道以後當「距離感測器」的數值是多少，「馬達轉速」應該設定為多少，進而實現自動駕駛。
  - 其中「距離感測器」的數值通常稱為「feature」，「馬達轉速」的數值通常稱為「target」



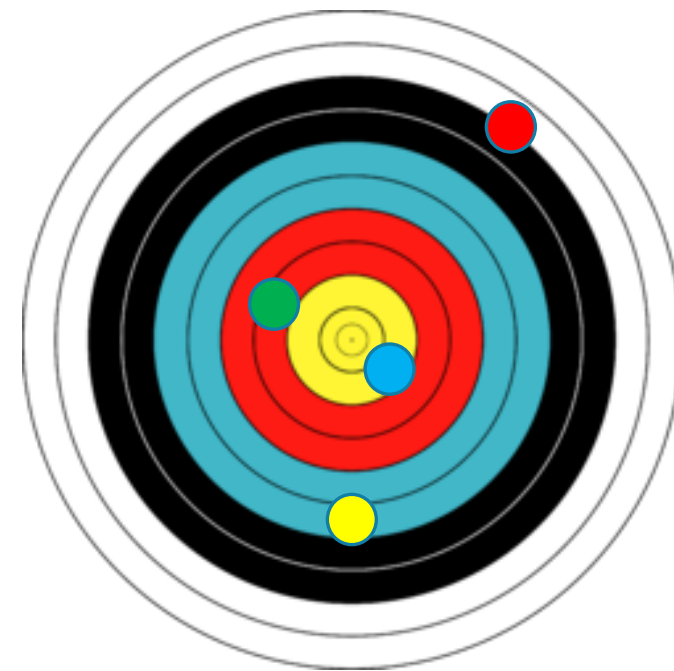
# 人工智慧簡介

- 在電腦能根據我們提供的資料集做出判斷之前，我們都要先使用這些資料集進行「學習」
- 所謂的「學習」，就是調整各種參數，讓feature的值經過運算之後，越接近我們給的target越好
- 學習出來的整個「參數的架構」，就稱為「模型」



# 人工智慧簡介

- 所謂的「學習」，就是調整各種參數，讓feature的值經過運算之後，越接近我們給的target越好
- 像這種有目標的學習方式，稱為「監督式學習」





# 人工智慧簡介

- 電腦學習的方式有很多種，通常我們也會說是不同的模型，例如
  - KNN：K個最近鄰居
  - SVM：線性支持向量機
  - DT：決策樹
  - MLP：多層感知器，等等，還有很多。



# 人工智慧簡介

- 不同的模型，就會有不同的「參數架構」
- 而每種模型適合的狀況不一樣，所以AI比賽，有一部份就是在比誰能挑到適合的模型





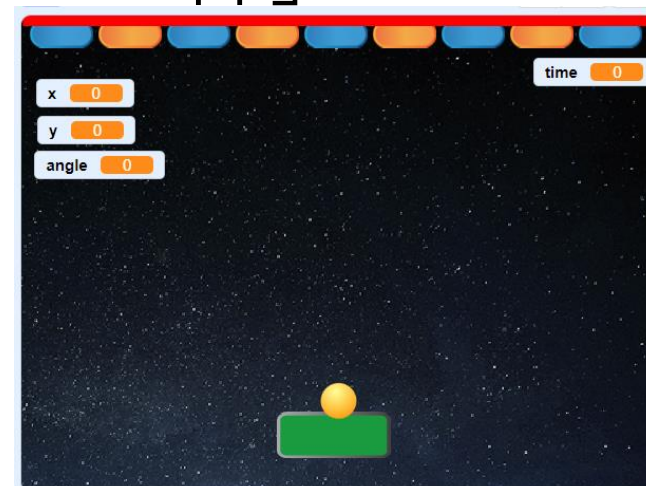
# 人工智慧簡介

- 除了有不同種類的學習模型之外，機器學習還可依照最後幫我們做的預測跟判斷，分為：
  - 迴歸模型：預測出來的結果是連續的數值的
  - 分類模型：預測出來的結果是某一種類別的



# 人工智慧簡介

- 以打磚塊遊戲為例，可以將橫桿的位置分成：
  - 迴歸模型：預測出來的結果是連續的X軸數值
  - 分類模型：預測出來的結果是某一種類別，例如「左」、「左中」、「中」、「中右」、「右」
- 然後橫桿跑到預測的位置去接球





# 人工智慧簡介

## ■ 題外話：

- 在預設程式中，迷宮自走車的輸出：「左右馬達的轉速」是連續的數值
- 你覺得用「分類」還是用「迴歸」模型會好一點？



# 人工智慧簡介

## ■ AI訓練的基本步驟

1. 收集資料：收集大量資料，分為訓練集和測試集。
2. 過濾資料：過濾出有特色的特徵 (影像則是由模型自動過濾)。
3. 挑選適合資料的學習模型：監督式、非監督式，分類、迴歸、預測...
4. 訓練模型：GPU利於平行運算、CPU利於循序運算。
5. 測試模型：利用測試集進行模型評估。
6. 應用模型：將訓練好的模型運用於各方面。



# 人工智慧簡介

## ■ 人工智慧的課程，到底要讓孩子學到什麼？

■ 基本原理？  
快速攻略（初階）

■ 大致瞭解AI在做什麼？

■ 調參數、換模型？

■ 選擇適合的資料？

■ 程式設計？

快速攻略（中階）

■ 各種演算法？

■ 過濾資料？

■ Python程式語言的用法

■ 生活應用？

高手養成  
（高階）

A background graphic featuring a complex network of interconnected nodes and lines. The nodes are represented by circles of various sizes and colors, including dark blue, light blue, and grey. The lines are thin and grey, creating a web-like structure. Some nodes are highlighted with larger, concentric circles. The overall aesthetic is modern and technological.

# BLOCKLY APP 用法

---

競賽攻略

# 競賽攻略-BLOCKLY APP用法



■ 軟體下載:PAIA網站→教材學習→Blockly APP

PAIA

遊戲學習 AI競技場 **教材學習** 捐款支持 進入PAIA

1. PAIA  
2. 開始遊戲  
3. 遊戲  
4. AI 競技場  
5. 教材  
**6. Blockly-App**  
7. C語言模組

PAIA

Paia = Playful AI Arena

教材大綱 ▼

- Paia = Playful AI Arena
- 平台架構

我們希望透過遊戲式學習的方式，吸引學生學習基礎程式語法、程式邏輯、機器學習等相關能力。最後達成在遊戲中學習、成長、應用等三大能力。



# 競賽攻略-BLOCKLY APP用法

## ■ 點選Github頁面

遊戲學習 AI競技場 教材學習

1. PAIA ✓

2. 開始遊戲

3. 遊戲

4. A I 競技場

5. 教材

6. Blockly-App ✓

7. C語言模組

PAIA > Blockly-App

Blockly-App

單機版積木程式 Blockly-App

程式安裝

進到 **Github 頁面** 後，可以在底下 Downloads 的位置找到最新版的 Windows 與 macOS 安裝檔，下載後執行就可以啟動程式了。

教材大綱 ▼

- 單機版積木程式 Blockly-App
  - ■ 程式安裝
    - 操作介面
    - 資料存取位置設定
- 模型積木
  - ■ 模型建立
    - 模型訓練
    - 模型預測
- 檔案積木
  - ■ 檔案儲存
- 自走車遊戲範例
  - ■ 以規則取向搜集訓練資料
  - 訓練機器學習遊戲模型





# 競賽攻略-BLOCKLY APP用法

■ 軟體裝好、執行遊戲、選擇迷宮自走車

■ 資料夾架構：

■ blockly-app\app-1.2.5\resources\app

\MLGame才是核心

■ 目前是1.2.5版

css	2021/9/10 下午 08:15	檔案資料夾
html	2021/9/10 下午 08:15	檔案資料夾
js	2021/9/10 下午 08:15	檔案資料夾
media	2021/9/10 下午 08:15	檔案資料夾
MLGame	2021/9/10 下午 08:15	檔案資料夾
node_modules	2021/9/10 下午 08:15	檔案資料夾
python	2021/9/10 下午 08:15	檔案資料夾
tutorials	2021/9/10 下午 08:15	檔案資料夾
xml	2021/9/10 下午 08:15	檔案資料夾
.gitmodules	2021/9/10 下午 08:15	文字文件
code	2021/9/10 下午 08:15	Chrome HTML D...
index	2021/9/10 下午 08:15	Chrome HTML D...
main	2021/9/10 下午 08:15	JavaScript 指令檔
package	2021/9/10 下午 08:15	JSON File
preload	2021/9/10 下午 08:15	JavaScript 指令檔
python	2021/9/10 下午 08:15	Chrome HTML D...
README.md	2021/9/10 下午 08:15	MD 檔案
renderer	2021/9/10 下午 08:15	JavaScript 指令檔
requirements	2021/9/10 下午 08:15	文字文件

# 競賽攻略-BLOCKLY APP用法



## ■ 資料夾架構：

■ 遊戲都放在MLGame\games下：

迷宮自走車在這裡

■ arkanoid

■ easy\_game

■ Maze\_Car

■ pingpong

■ RacingCar

■ snake

# 競賽攻略-BLOCKLY APP用法



## ■ 資料夾架構：

■ **ml**資料夾裡是feature、target跟模型等資料

■ **src\map**資料夾裡是地圖

■ 「開啟檔案存取路徑」會到ml資料夾裡

名稱

.idea

asset

game\_test

ml

src

\_\_init\_\_.py

blockly.json

config.py

game\_config.json

main.py

README.md

README\_blockly.md

README\_python.md

requirements.txt

機器學習資料  
內有地圖



# 競賽攻略-BLOCKLY APP用法

## ■ 範例程式：

■ 1.Start：初始的基本架構

■ 2.Manual：手動蒐集資料

■ 3.Model\_training：訓練模型

■ 4.Model\_playing：依照模型自動駕駛

範例程式 ▾

語言 ▾

選項 ▾

1. start.xml

2. manual.xml

3. model\_training.xml

4. model\_playing.xml



# 競賽攻略-BLOCKLY APP用法

- 對初階的學生來說，依序使用這三個範例程式就好：
  - 2.Manual：手動蒐集資料
  - 3.Model\_training：訓練模型
  - 4.Model\_playing：依照模型自動駕駛



# 競賽攻略-BLOCKLY APP用法

- 對中階的學生來說，使用這兩個範例，製作出依照規則自動蒐集資料的機器人，短時間內取得大量的資料：
  - 1.Start:初始的基本架構
  - 2.Manual：手動蒐集資料



# 競賽攻略-BLOCKLY APP用法

- 對高階的學生則是可以：
  - 使用python將資料倒出來檢查，並經過處理，將雜訊過濾
  - 深入研究各種模型適合的狀況以及其原因



# 競賽攻略-BLOCKLY APP用法

- 既然「人工智慧是電腦根據我們提供的資料集，做出判斷」
- 那資料集就要夠大、資料要夠多，AI才會學到各種狀況，才會厲害
- 但內建地圖數量不多（5+3，而且太簡單），所以建議要自己建立地圖來蒐集資料
- 所有資料皆為比賽當場蒐集，所以要教學生自己建地圖





# 競賽攻略-BLOCKLY APP用法

- 訓練模型時，是選「執行python檔」
- 要手動開車或自動駕駛時，是選「玩遊戲」
- 每秒顯示張數
- 參賽人數
- 模式
- 地圖
- 遊戲時間
- 距離感測器個數
- 聲音





# 競賽攻略-BLOCKLY APP用法

- 每秒顯示張數（FPS）：
  - 其他遊戲的速度是由FPS決定的，FPS越高，球跑得越快。
  - 迷宮自走車的速度是由另外的參數控制，所以FPS單純只影響每秒蒐集到的資料量。
  - 至於最好的FPS要設定多少？就要自己測試了



# 競賽攻略-BLOCKLY APP用法

## ■ 參賽人數：

- 使用手動收集資料時，沒有影響，因為按下方向鍵控制車子時，所有的車子會同步反應，所以都疊在一起
- 使用規則讓車子自動蒐集資料時，或許就有差，如果積木裡有隨機的部分，那每台車反應不同，可同時蒐集到更多資料。
- 使用AI自駕時，也會因為細微的差異，導致每台車走的不太一樣，可看到自己跟自己比賽。



# 競賽攻略-BLOCKLY APP用法

- 模式：其實只是不同的地圖
  - 小試身手對應：level\_xxx
  - 經典迷宮對應：normal\_map\_xxx
  - 移動迷宮對應：move\_map\_xxx
- 地圖：就是檔名的編號

- 🔊 level\_1
- 🔊 level\_2
- 🔊 level\_3
- 🔊 level\_4
- 🔊 level\_5
- 🔊 level\_6
- 🔊 level\_7
- 🔊 level\_8
- 🔊 level\_9
- 🔊 level\_10
- 📄 map\_test
- 🔊 move\_map\_1
- 🔊 move\_map\_2
- 🔊 move\_map\_3
- 🔊 move\_map\_4
- 🔊 normal\_map\_1
- 🔊 normal\_map\_2
- 🔊 normal\_map\_3
- 🔊 normal\_map\_4
- 🔊 normal\_map\_5



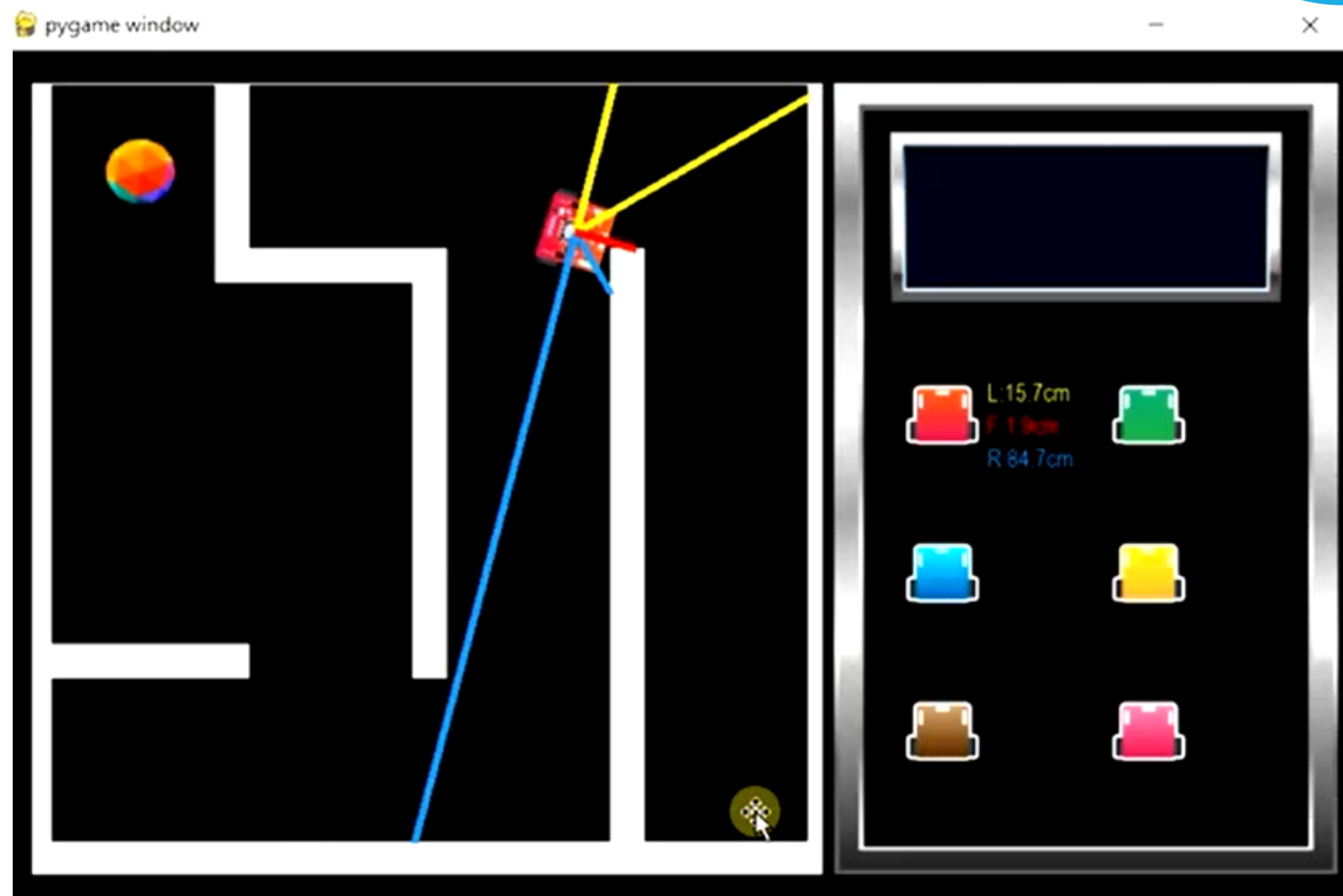
# 競賽攻略-BLOCKLY APP用法

- 遊戲時間：經過多少frame時停止遊戲
- 距離感測器：建議用5個，蒐集到較多資訊
- 聲音：可以關掉

# 競賽攻略-BLOCKLY APP用法



■ 遊戲畫面：



# 競賽攻略-BLOCKLY APP用法



- 地圖視野會隨著車子的位置變化，但也可以用「IKJL」上下左右移動視野看地圖



# 競賽攻略-BLOCKLY APP用法



■ 製作好的程式可以存為XML檔

■ 比賽的網站只吃python檔，所以要用網站時，要點這個

■ 這個可以連到訓練資料的資料夾





The background of the slide features a complex network diagram. It consists of numerous nodes of varying sizes, colored in dark blue, light blue, and grey. These nodes are interconnected by a web of thin, light grey lines, creating a sense of connectivity and data flow. Some nodes are highlighted with larger, concentric circles, suggesting key points or hubs within the network.

# PAIA網站相關功能

---

競賽攻略



# 競賽攻略-PAIA網站相關功能

- 登入網站後，在「AI競技場」→「可報名」，點這裡：

AI 競技場

可報名


已參加

可管理

點此輸入代碼，快速加入比賽

PTWA測試用競賽

報名與上傳時間:  
2021-01-27 16:46:50 ~ 2099-01-27 16:46:51  
比賽時間: 2099-01-27 16:47:01  
比賽項目: Maze\_Car  
用於加入競賽之教學



開放報名中



# 競賽攻略-PAIA網站相關功能

■ 選「上傳新增」

可選擇程式					
編號	A I 的名字	說明	狀態	使用積木	上傳新增
1	 Untitled		待命中 <span>出賽</span>	  	
2	 Alcar0912	Alcar0912	待命中 <span>出賽</span>	  	
3	 Al-car1		待命中 <span>出賽</span>	  	
4	 Alcar1	Al-car1	待命中 <span>出賽</span>	  	



# 競賽攻略-PAIA網站相關功能

迷宮自走車

A I 的名字

填寫

請使用英文與數字

說明

選擇檔案

未選擇任何檔案

選擇檔案。注意！一次要選擇兩個檔案：  
AI程式，以及模型檔。  
其中AI程式的名稱必須為：ml\_play.py

點擊這裡選擇或刪除文件

關閉

提交



# 競賽攻略-PAIA網站相關功能

■ 上傳完畢後，點「播放」，就可以運作了

可選擇程式					
編號	A I 的名字	說明	狀態	<a href="#">使用積木</a> <a href="#">上傳新增</a>	
1	 Untitled		待命中 <a href="#">出賽</a>	  	
2	 Alcar0912	Alcar0912	待命中 <a href="#">出賽</a>	  	
3	 Al-car1		待命中 <a href="#">出賽</a>	  	
4	 Alcar1	Al-car1	待命中 <a href="#">出賽</a>	  	



# 競賽攻略-PAIA網站相關功能

- 若沒有「ml\_play.py」，頁面會報出錯誤訊息
- 若沒有「model.pickle」，則會停留在白色畫面



# 競賽攻略-PAIA網站相關功能

■ 練習時，其實選「遊戲學習」也可以





# 競賽攻略-PAIA網站相關功能

- 比賽時，兩個學生共用一個帳號
- 不同關卡，可以使用不同的AI出戰
  - 所以可以針對關卡繪製地圖來訓練，可以提高分數
  - 但決賽是最後才看的到地圖喔





# 自行製作地圖

---

競賽攻略



# 競賽攻略-自行製作地圖

- 既然「人工智慧是電腦根據我們提供的資料集，做出判斷」
- 那資料集就要夠大、資料要夠多，AI才會學到各種狀況，才會厲害
- 但內建地圖數量不多（5+3，而且太簡單），所以要自己建立地圖來蒐集資料



# 競賽攻略-自行製作地圖

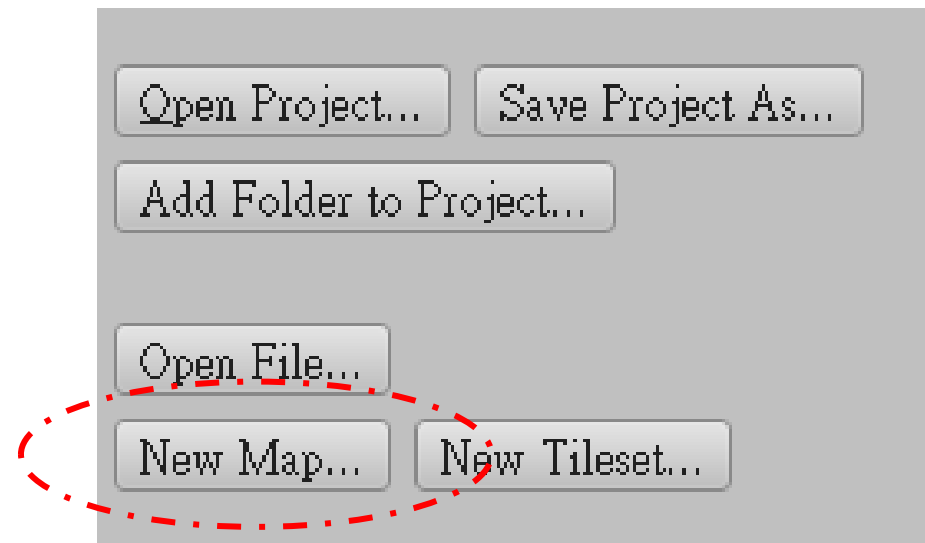
- 使用「Tiled-1.7.1-win64」程式

- <https://www.mapeditor.org/download.html>

- 第一次開啟，選「new map」

- Tileset中文是圖塊集，

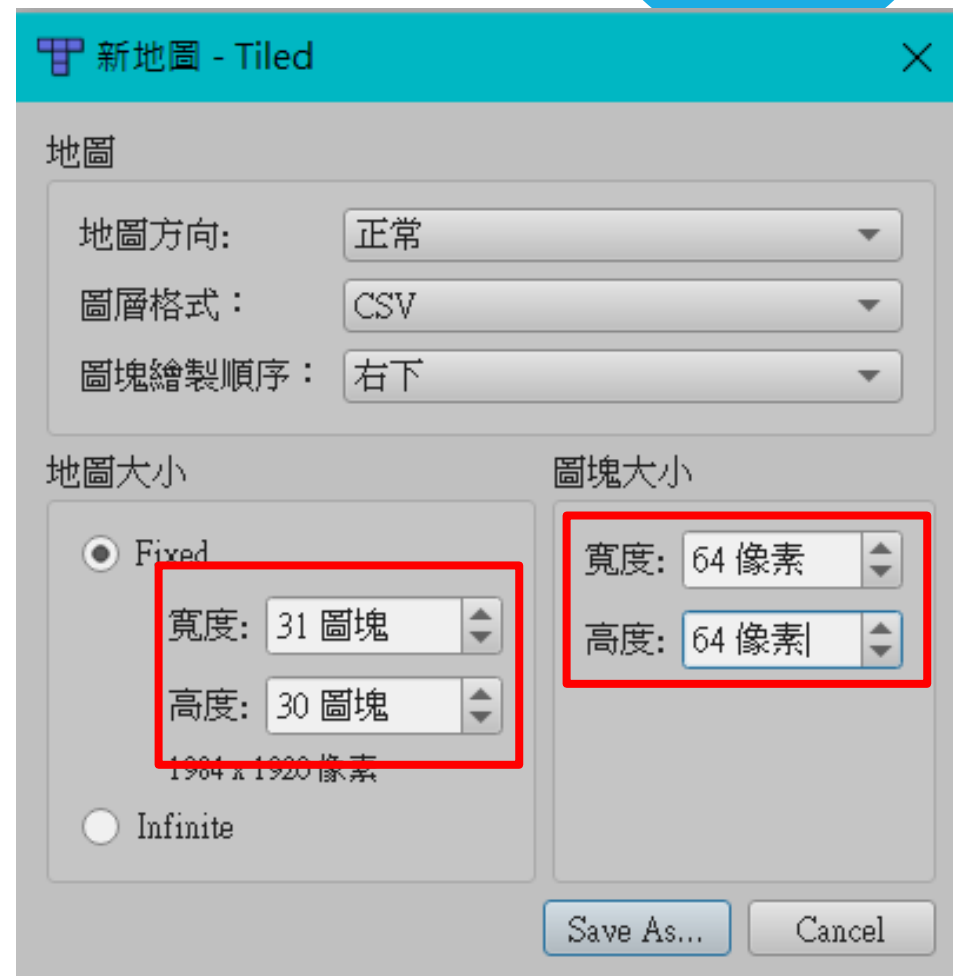
就是這張





# 競賽攻略-自行製作地圖

- 兩個參數：
- 地圖大小：（比賽規定）
  - 預賽為30x30圖塊（決賽40X40）
  - 但請設定為31x30圖塊
- 圖塊大小：
  - 設定為64x64像素（by這張圖的像素）





# 競賽攻略-自行製作地圖

- 檔名請照此規則：
  - 小試身手：level\_11.json
  - 經點迷宮：normal\_map\_6.json
  - 注意！請用json檔

檔案名稱(N): normal\_map\_5

存檔類型(T): JSON map files (\*.json)



# 競賽攻略-自行製作地圖

## ■ 放置位置：

■ C:\Users\circl\AppData\Local\blockly-app\app-1.2.4\resources\app\MLGame\games\Maze\_Car\src\map

■ Blockly app 「開啟檔案路徑」後，「上一層」就會看到「src」了

- level\_1
- level\_2
- level\_3
- level\_4
- level\_5
- level\_6
- level\_7
- level\_8
- level\_9
- level\_10
- map\_test
- move\_map\_1
- move\_map\_2
- move\_map\_3
- move\_map\_4
- normal\_map\_1
- normal\_map\_2
- normal\_map\_3
- normal\_map\_4
- normal\_map\_5

# 競賽攻略-自行製作地圖

- 這裡要放入製作地圖的圖塊 ( Tileset )

■ 也就是這張：

- Tiled軟體，會依照你剛剛設定的像素來切格子

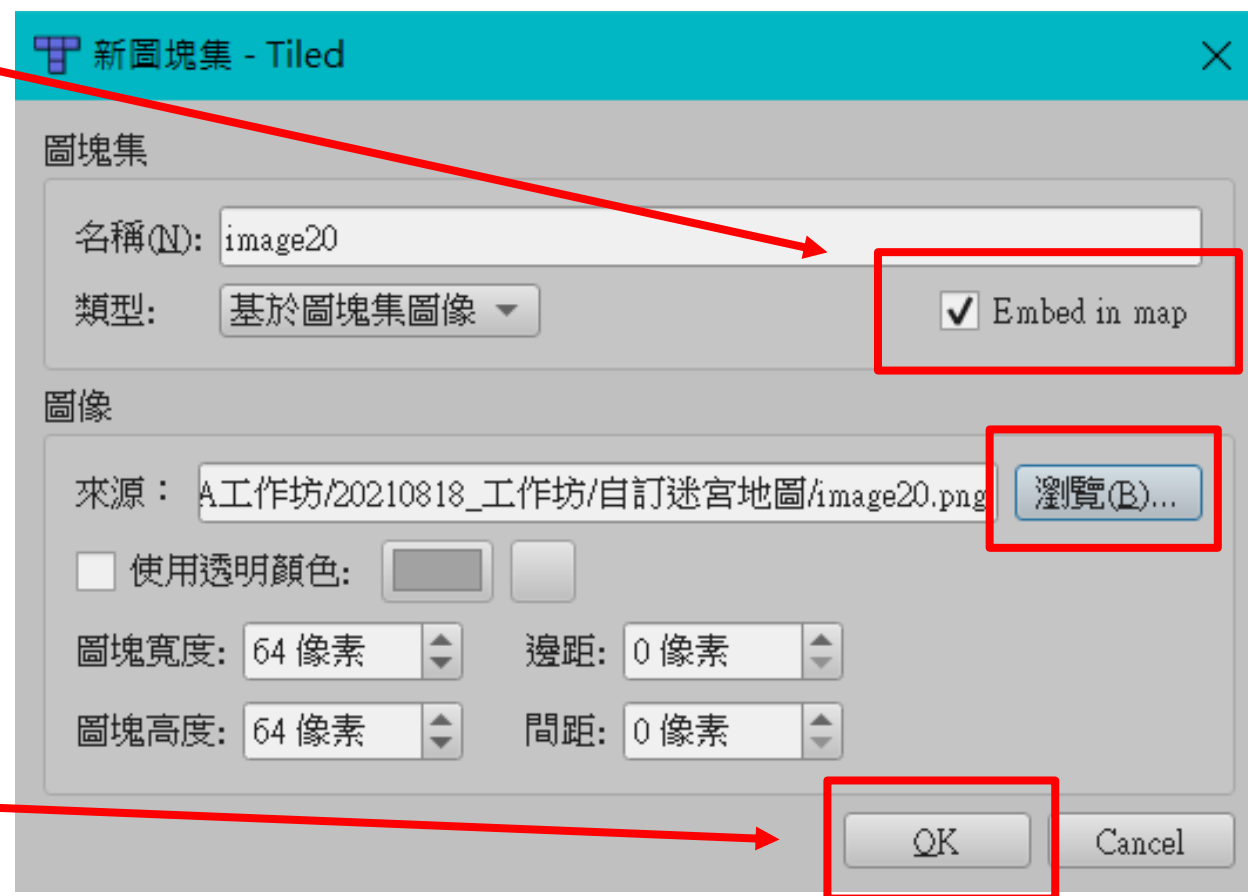
- 因為這張圖片每格都是64像素，正方形所以剛剛要設定64\*64





# 競賽攻略-自行製作地圖

- 這個記得勾選
- 名稱不用填，選好檔案之後會自動帶入
- 沒有勾，這裡會變成「save as」



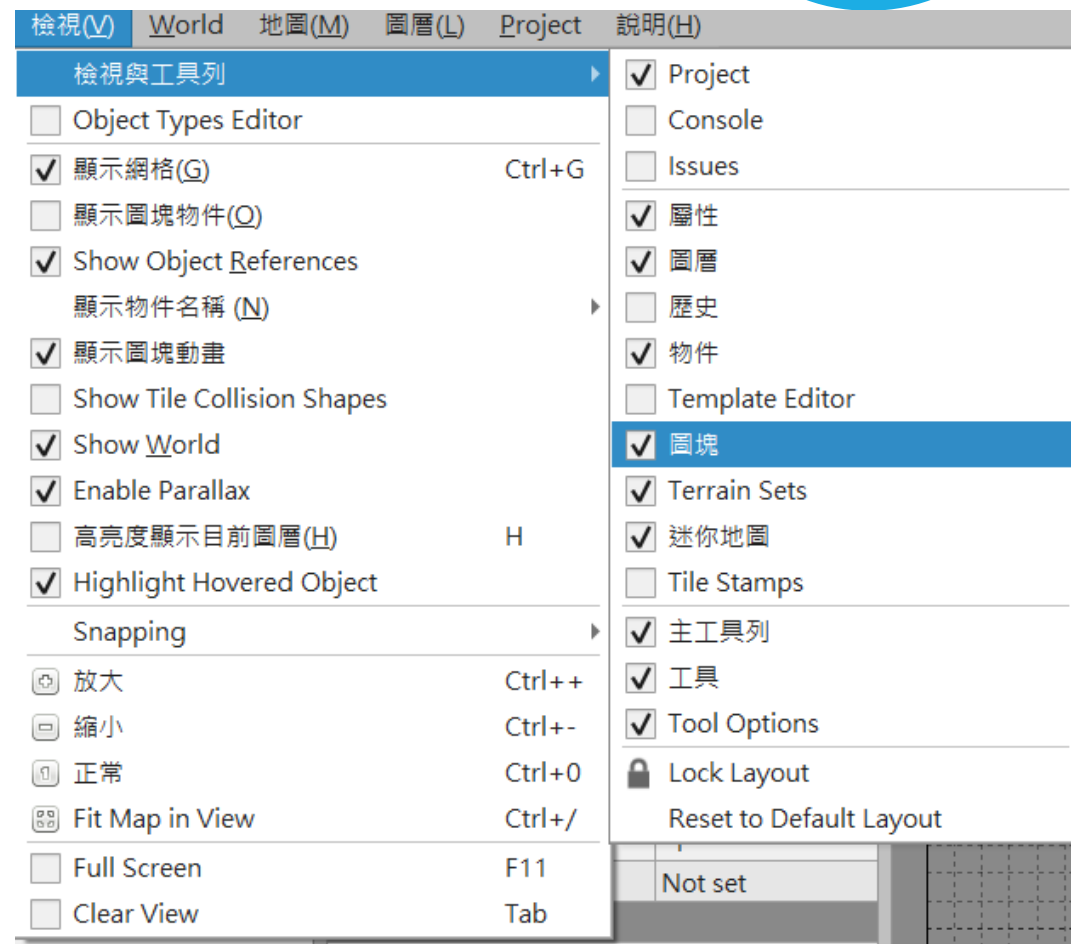




# 競賽攻略-自行製作地圖

■ 如果圖塊檢視區塊不小心被關掉了，請去：

■ 檢視→檢視與工具列→圖塊打勾





# 競賽攻略-自行製作地圖

## ■ 圖塊：

- 使用牆壁、起點跟終點





# 競賽攻略-自行製作地圖

- 選好圖塊後，使用工具列：
  - 可用「印章」一格一格慢慢蓋
  - 或用「矩形選取」框一個區塊，用「油漆桶」倒滿
    - 不用了記得取消
  - 也可直接用「矩形填滿」
  - 「橡皮擦」可用來擦掉





# 競賽攻略-自行製作地圖

## ■ 特別注意！

- 右牆必須蓋兩層（所以寬度要選31格）
- 一定要有起點跟終點，否則會出錯
- 車子圖示是真正車子的上端，繪製出來的車子大小約2格多一點，所以要留3格以上的空間讓車子過
- 終點圖示是真正終點的上端，所以左、右、下方要離牆壁3格



# 運算思維

---

競賽攻略



# 競賽攻略-運算思維

## ■ 預設範例中：

- 使用方向鍵或WSAD控制車子移動

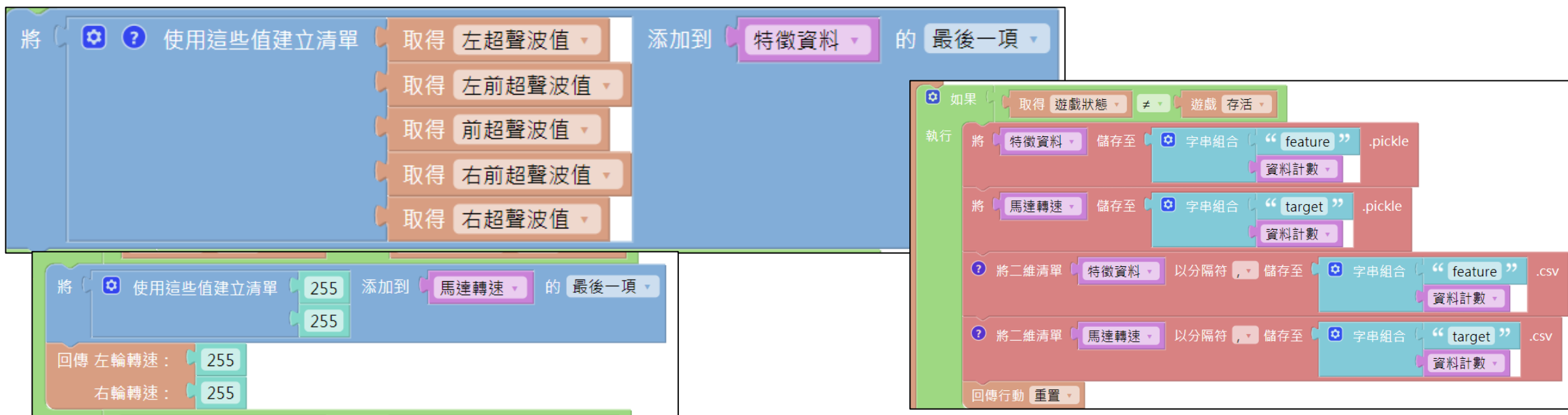
- 注意！方向不是絕對的上下左右，而是車子的前後左右





# 競賽攻略-運算思維

- 通常是把特徵（feature）先做成清單存到變數（遊戲內部）裡，遊戲結束後再存成pickle檔（外部檔案）。

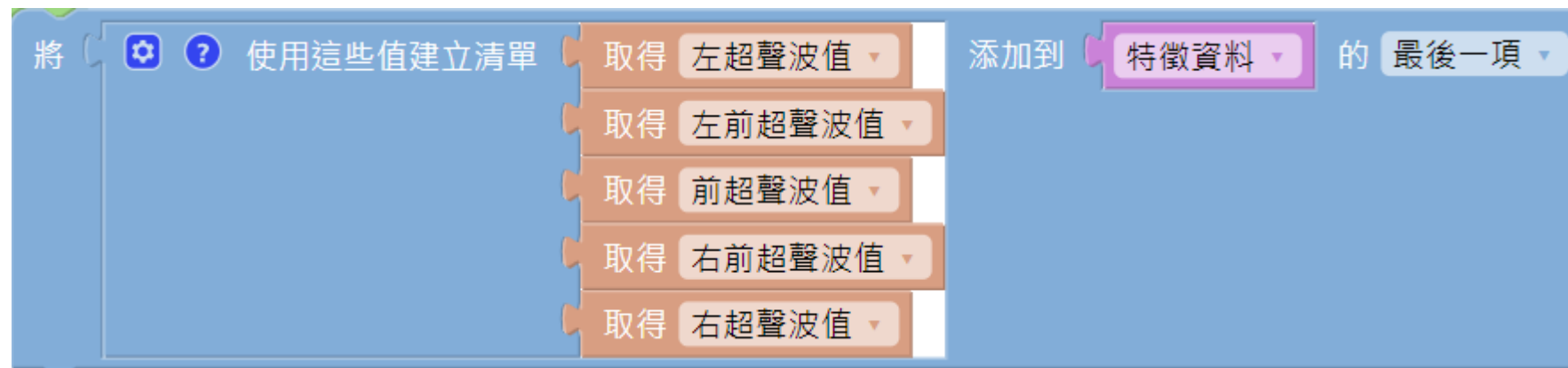




# 競賽攻略-運算思維

## ■ 預設範例中：

- 抓取5個距離感測器的數值作為「feature」
- 抓取2個馬達轉速的數值作為「target」（上一張投影片）

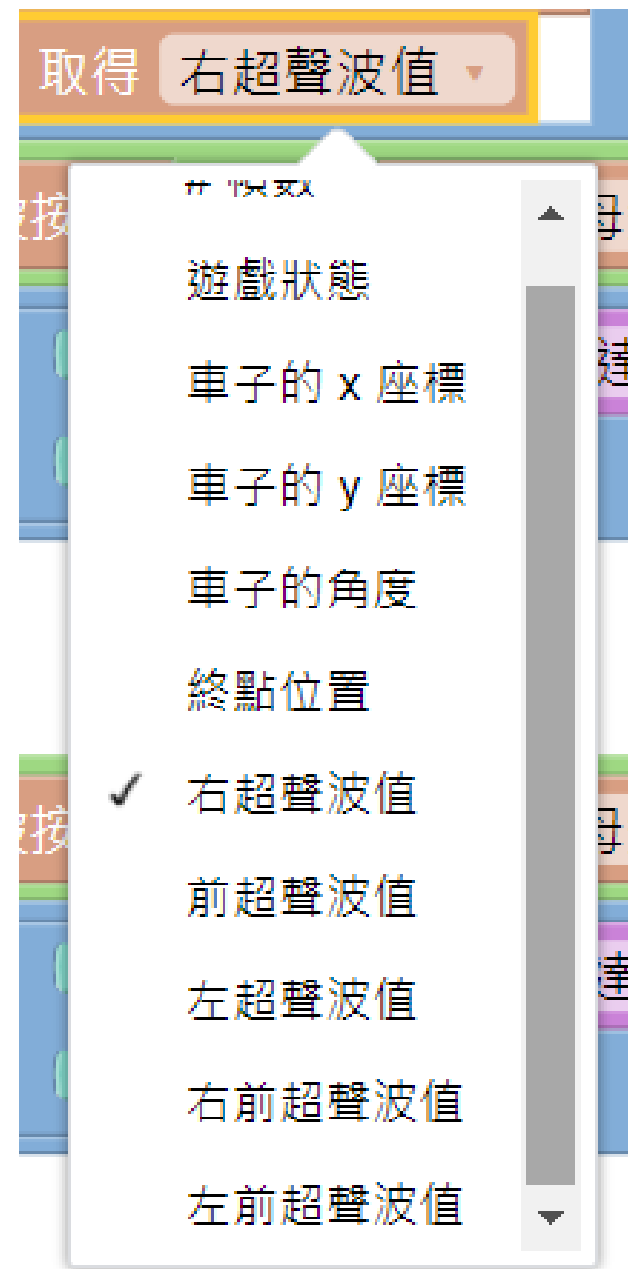






# 競賽攻略-運算思維

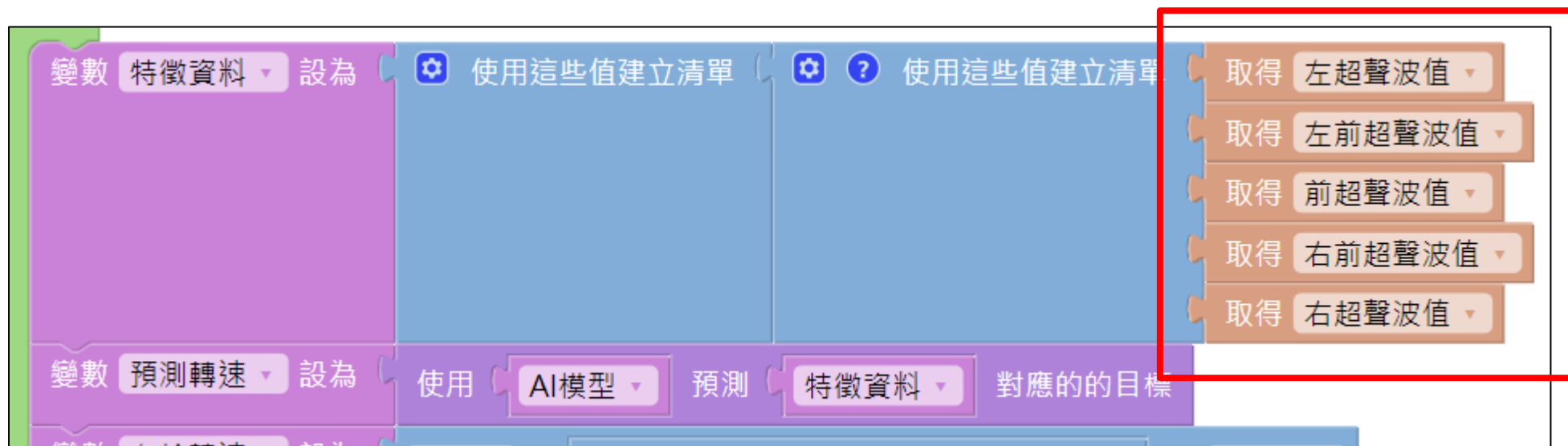
- 可不可以抓更多或更少呢？
- 當然可以！有這些可以抓：





# 競賽攻略-運算思維

- 但如果有調整，在範例程式 4.model\_playing.xml，就要有對應的修改：





# 競賽攻略-運算思維

## ■ 預設範例中：

- 遊戲結束一次（到達終點或時間結束），會產生一對紀錄。（意即提早關掉是不會有紀錄的）
- 每次「玩遊戲」，記錄編號會從1開始，覆蓋掉之前的。（多人合作要修改）





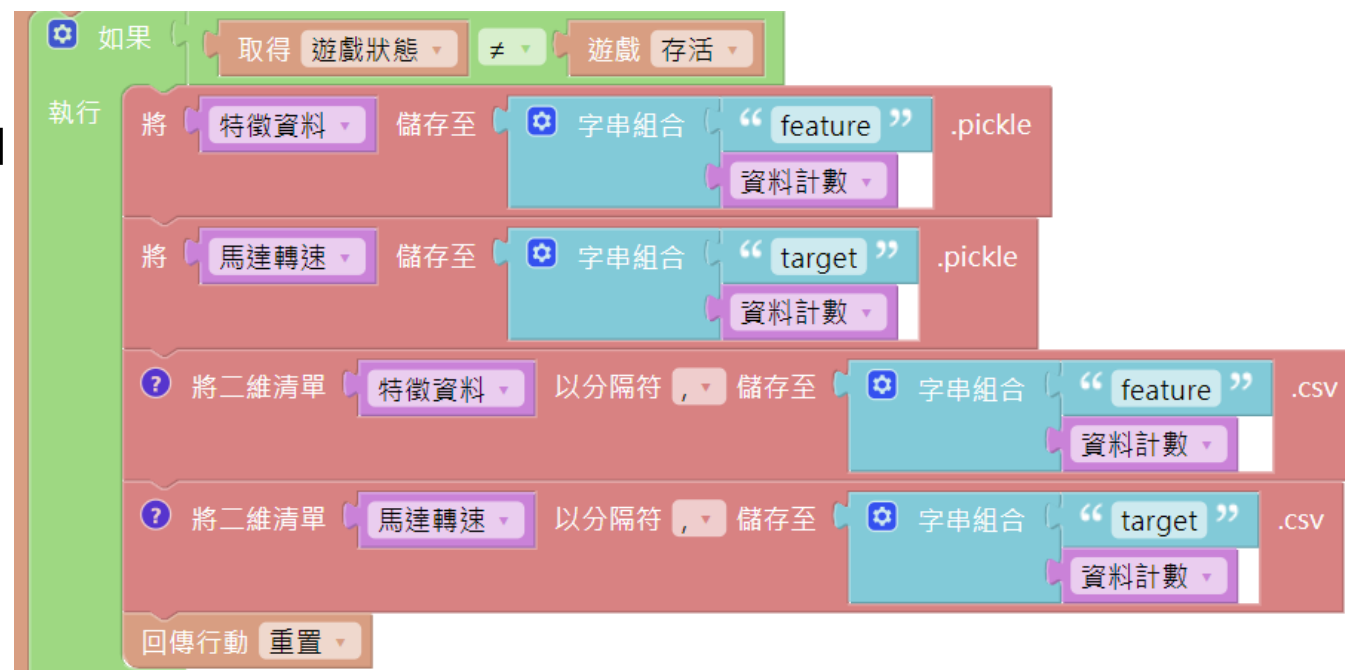
# 競賽攻略-運算思維

## ■ 記錄檔位置：

### ■ 「開啟檔案存取路徑」

■ or:  
C:\Users\circl\AppData\Local\blockly-app\app-1.2.4\resources\app\MLGame\games\Maze\_Car\ml

### ■ 有給人看的csv檔跟給電腦看的pickle檔

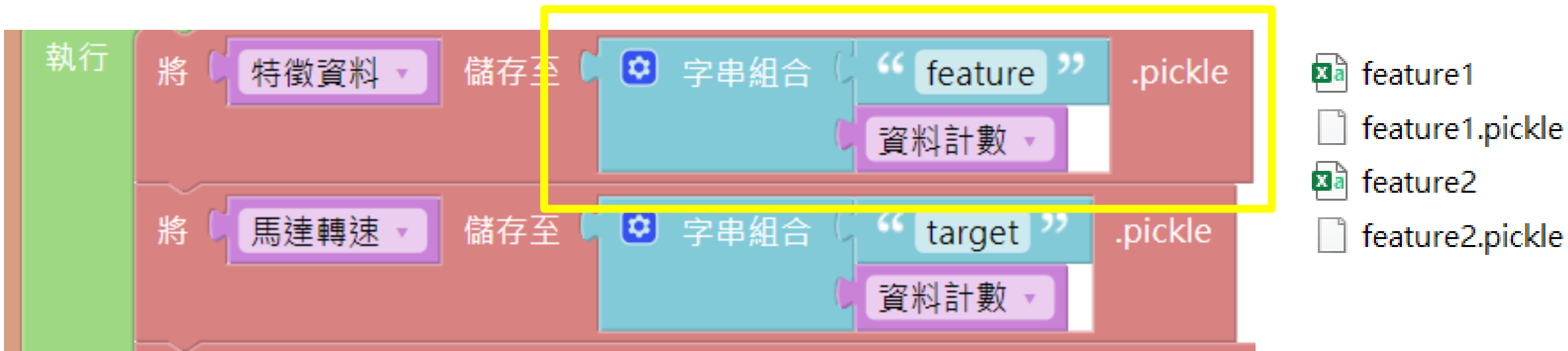




# 競賽攻略-運算思維

## ■ 特別說明：

- 用這樣的寫法，就會在記錄檔後面加上字串，一次玩遊戲，就會有多筆資料





# 競賽攻略-運算思維

- 因為遊戲的狀態有三種，所以目前範例程式的寫法，是「通關」或「失敗」都會記錄資料。
- 如果只想蒐集「通關」的資料，要自行修改

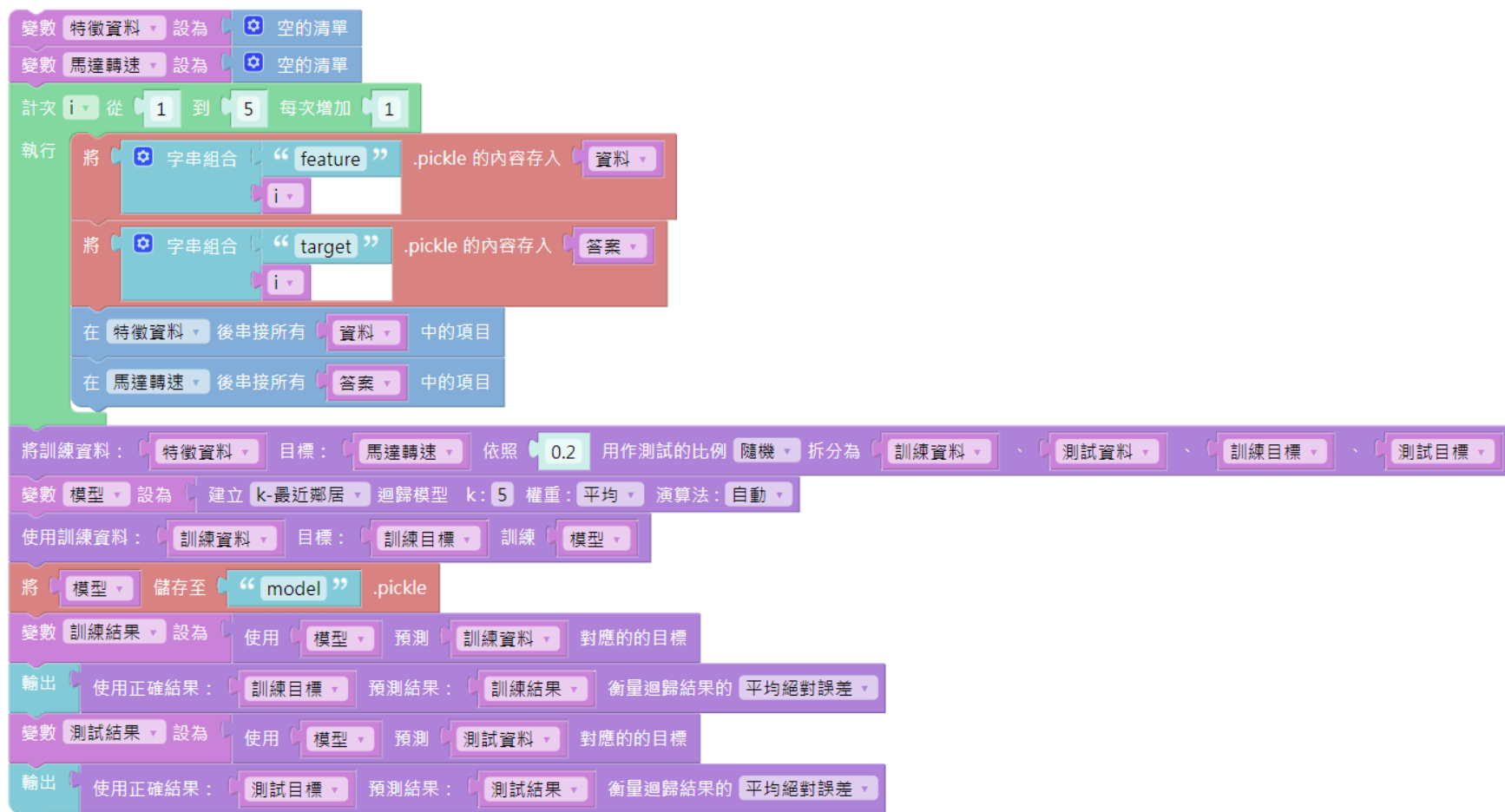




# 競賽攻略-運算思維

## ■ 訓練模型：

■ 分為三個部分：





# 競賽攻略-運算思維

## ■ 訓練模型：

- 處理資料的部分，這樣寫可以把很多筆 feature 放進「特徵資料」而不覆蓋舊資料







# 競賽攻略-運算思維

## ■ 訓練模型：

■ 把好幾筆記錄先整合為一筆

■ 這個數值要視情況修改





# 競賽攻略-運算思維

## ■ 訓練模型：

### ■ 訓練模型的主要區塊

用於看看錯誤率

The screenshot displays a machine learning workflow interface with the following components:

- 將訓練資料：** 特徵資料 (Feature Data) | 目標：馬達轉速 (Target: Motor Speed)
- 依照 0.2 用作測試的比例 隨機 拆分為：** 訓練資料 (Training Data) | 測試資料 (Testing Data) | 訓練目標 (Training Target) | 測試目標 (Testing Target)
- 變數 模型 設為：** 建立 k-最近鄰居 迴歸模型 (Build k-Nearest Neighbors Regression Model) | k: 5 | 權重：平均 (Weight: Average) | 演算法：自動 (Algorithm: Automatic)
- 使用訓練資料：** 訓練資料 (Training Data) | 目標：訓練目標 (Training Target) | 訓練 模型 (Train Model)
- 將 模型 儲存至：** “model” .pickle

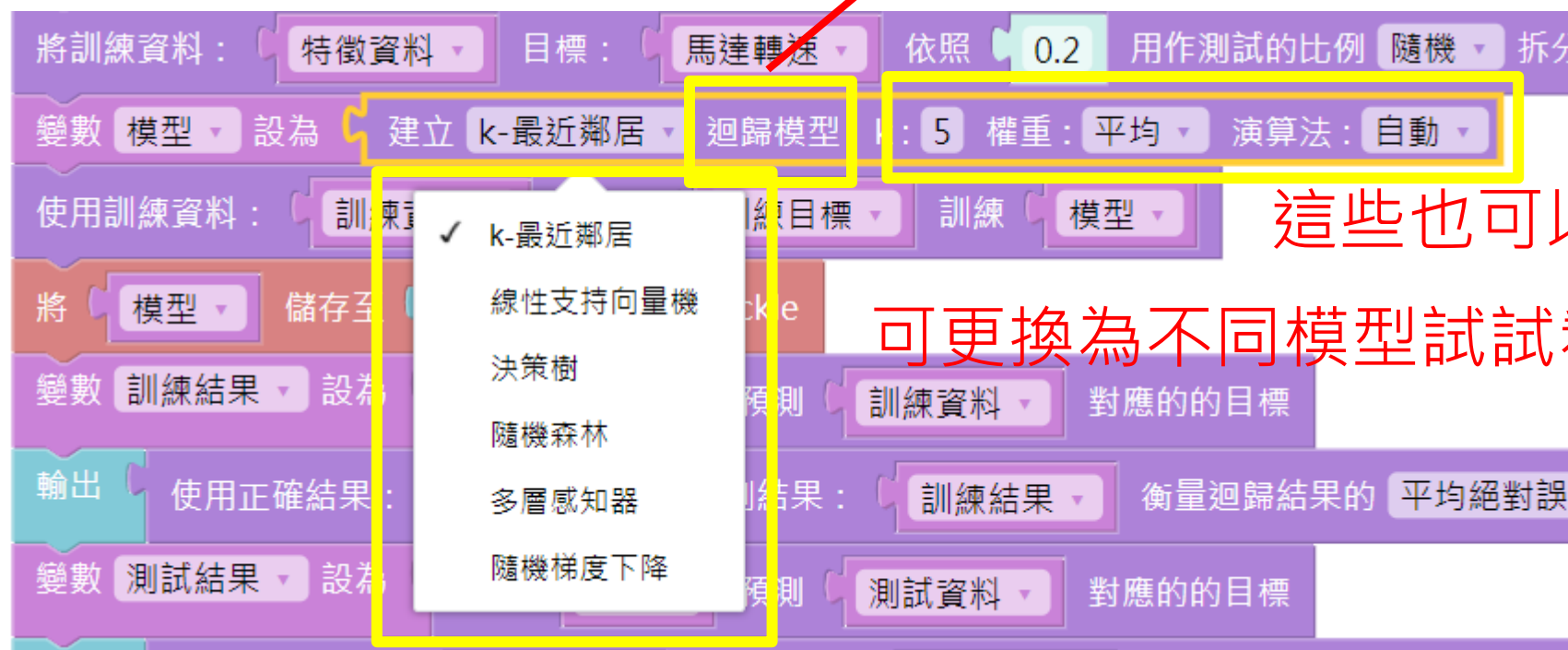


# 競賽攻略-運算思維

## ■ 訓練模型：

### ■ 訓練模型的主要部分

這也可以換為分類模型的積木試試看



這些也可以調整看看

可更換為不同模型試試看



# 競賽攻略-運算思維

## ■ 訓練模型：

■ 印出相關數值來看學習效果，作為參考





# 全自動收集資料

---

運算思維



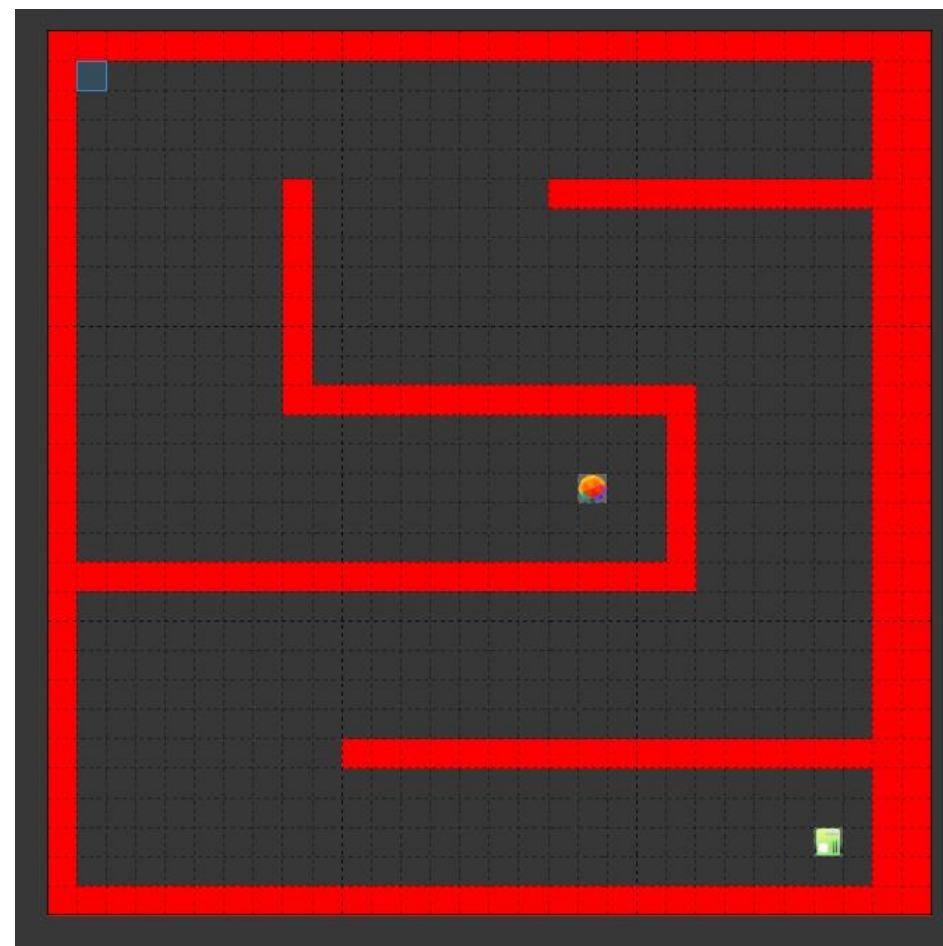
# 競賽攻略-運算思維

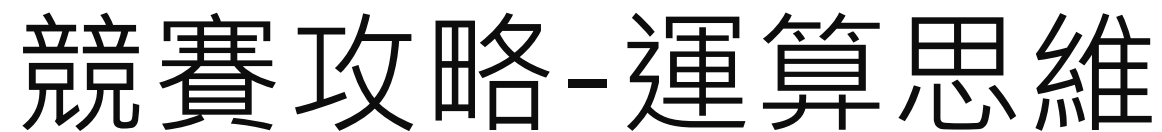
- 中階學生，製作自動收集資料的車子：
  - 延邊演算法
  - rule-base(DT)演算法
  - bfs演算法



# 競賽攻略-運算思維

- 延邊演算法：
  - 一直靠著某一邊，總有一天會走出去





■ 如果左邊有空間 → 向左偏轉

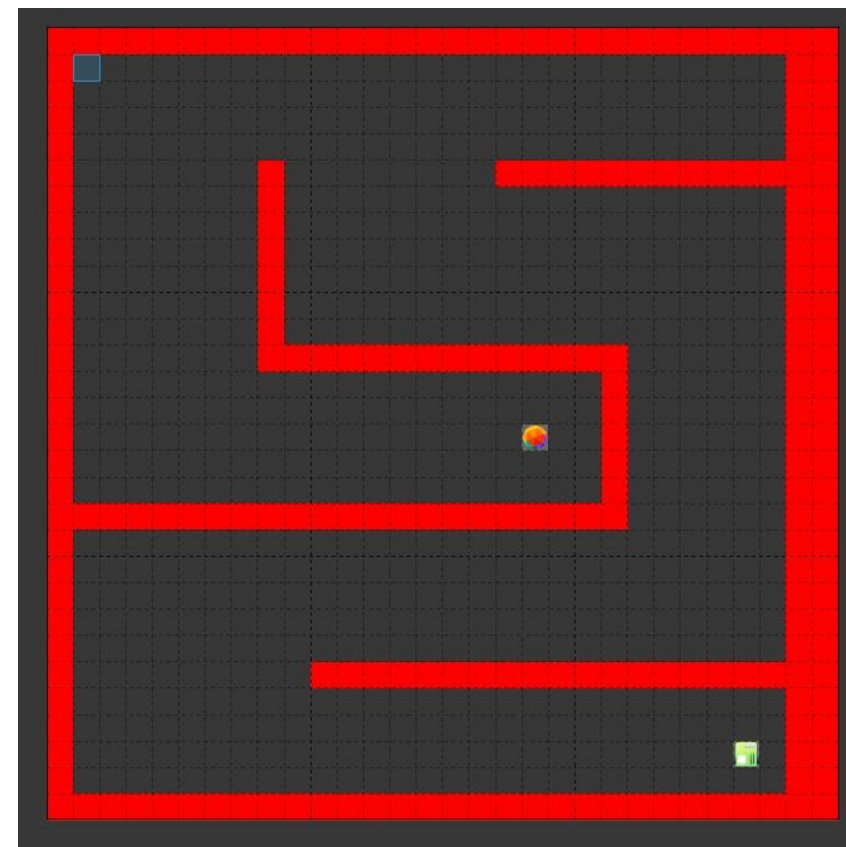
■ 否則如果 左邊、前面都沒有空間 → 右轉

■ 否則如果 死路(自行判斷)→ 迴轉

■ 否則如果 卡住→後退

■ 否則如果 前感應器 最大值 → 前行且左偏

■ 否則 緩慢前進







# 競賽攻略-運算思維

- 不過對於中階學生來說，並非寫出製作自動收集資料的車子就好，學生還要瞭解積木的基本用法，以及如何把想要的資訊建立為清單，然後存成變數，然後存到.pickle檔等等的功能。
- 如果不夠熟練，比賽時或許會花很多時間揪錯
- 寫不出自動沿邊程式怎麼辦？



# 教師待辦事項

---

結論



# 教師待辦事項

- 教學生blockly-APP的用法
  - 會開範例、會進入遊戲、會儲存等等
- 教學生繪製地圖，並放入正確的資料夾，讓遊戲抓的到。
  - 建議畫一張學生不用來做訓練的，作為測試AI效果用。
- 讓學生大量的玩遊戲，蒐集資料
  - 要對資料量、訓練時間等有概念，在比賽時，3小時內要完成。

初階

- 調整模型的參數，看看效果如何
- 加入更多的數值，看看效果如何
- 換不同的模型，看看效果如何
- 使用自動蒐集資料的機器人，蒐集到更多資料

中階

- 使用python來處理資料集，過濾雜訊

高階

The background of the slide is a complex network diagram. It features numerous nodes of varying sizes, some solid black, some solid blue, and some white with black outlines. These nodes are interconnected by a web of thin, light gray lines. The overall aesthetic is modern and technological, with a light gray background and subtle circular patterns.

# THNAK YOU

---

[oceanblue0212@hsjh.tn.edu.tw](mailto:oceanblue0212@hsjh.tn.edu.tw)