

國立臺北科技大學
2023 資工系物件導向程式實習
期末報告

瑪利貓



第 13 組

目錄

一、 簡介

1. 動機
2. 分工

二、 遊戲介紹

1. 遊戲說明
2. 遊戲圖形
3. 遊戲音效

三、 程式設計

1. 程式架構
2. 程式類別
3. 程式技術

四、 結語

1. 問題及解決方法
2. 時間表
3. 貢獻比例
4. 自我檢核表
5. 收穫
6. 心得或感想
7. 對於本課程的建議、想說的話

一、 簡介

1. 動機

瑪利貓是我們小時候玩的橫向跑庫遊戲，它可愛的畫風和卑鄙的陷阱深深吸引著我們，經常玩到非常生氣，但不服輸的精神讓我們從沒來有放棄通關它，不斷嘗試與努力之下，終於在我們大二時把它給通關了，為了讓助教也體會到這份成就感，我們決定來製作這款遊戲。

2. 分工

韓正勤(110590008)：物理 碰撞 部分陷阱 怪物

張塔信(110590039)：碰撞 部分陷阱 地圖 音效 其他

二、 遊戲介紹

1. 遊戲說明

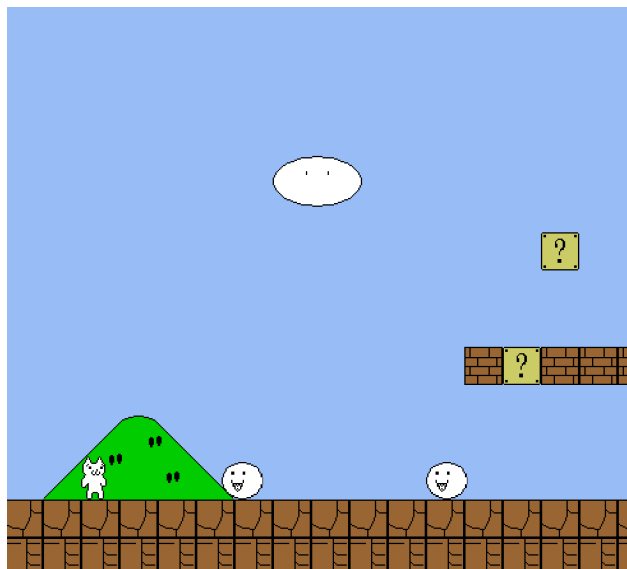
遊戲類型是橫向卷軸的跑庫遊戲，風格與瑪利歐非常相似，都是拿取道具、擊殺敵人和越過陷阱，最終抵達終點，不同的是這款遊戲的陷阱比較多，第一次玩的人需要死很多次才能慢慢推進。

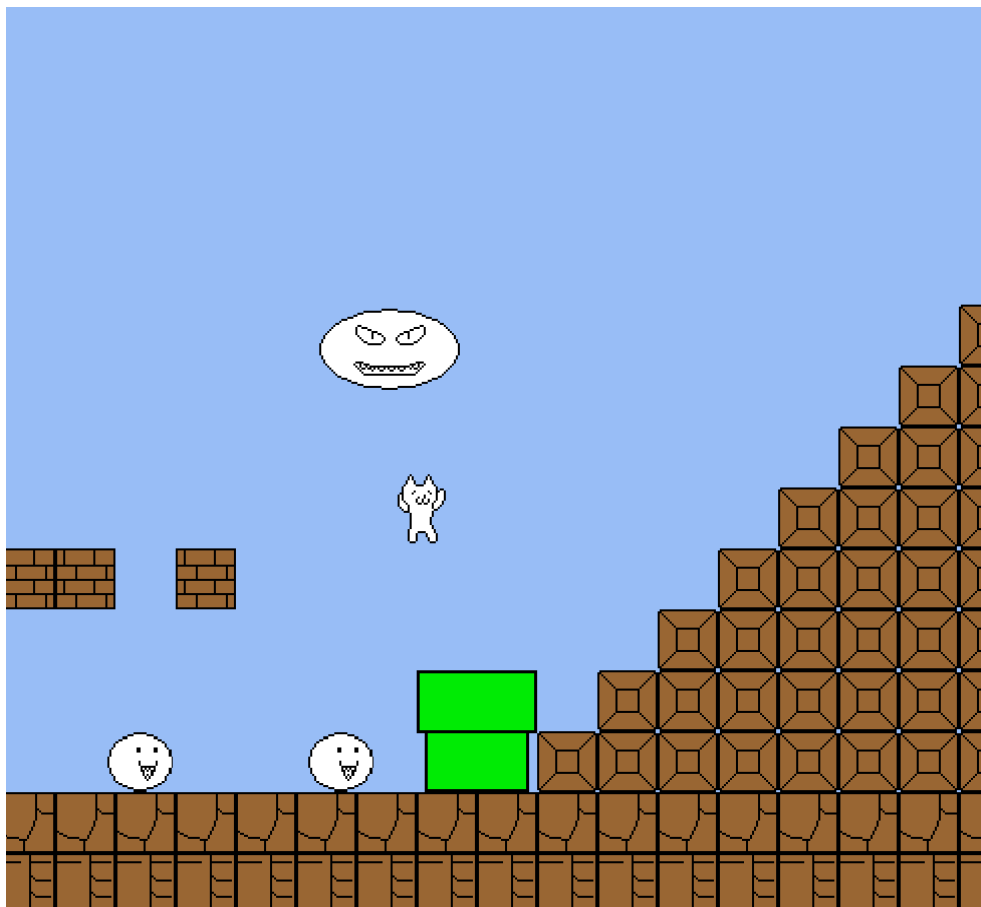
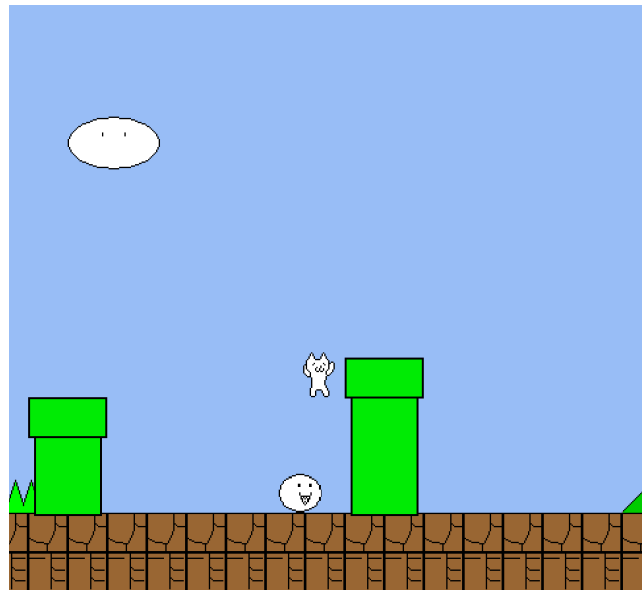
玩家需要用上下左右來控制瑪利貓的移動和跳躍。

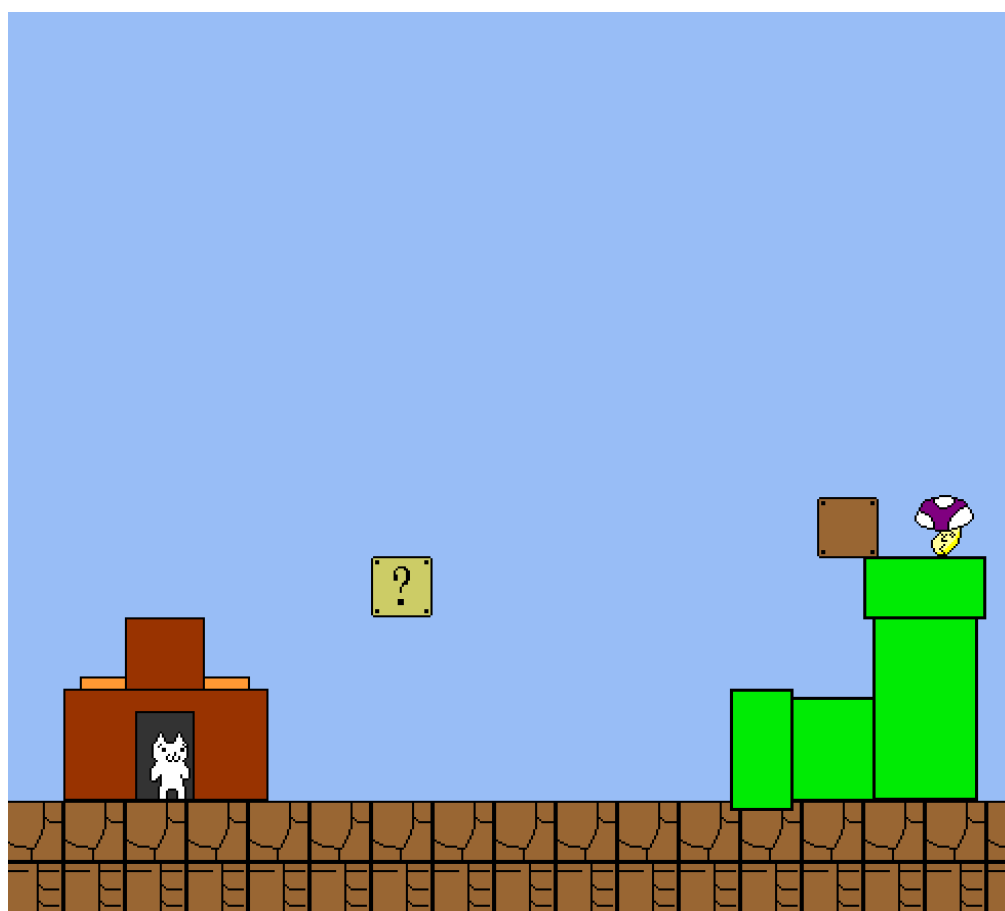
遊戲分為1、2兩個關卡，分別為地上和地下的地圖，每走個一兩步就會有意想不到的陷阱觸發，也有記錄點供玩家存檔。

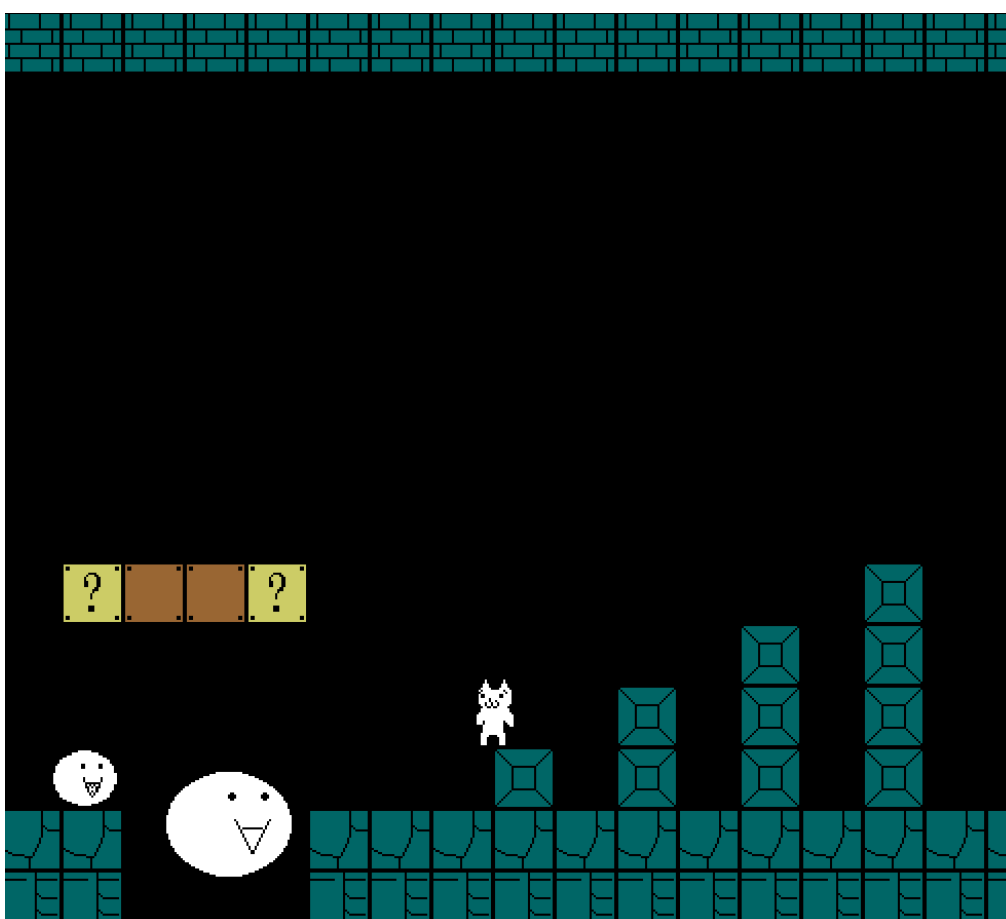
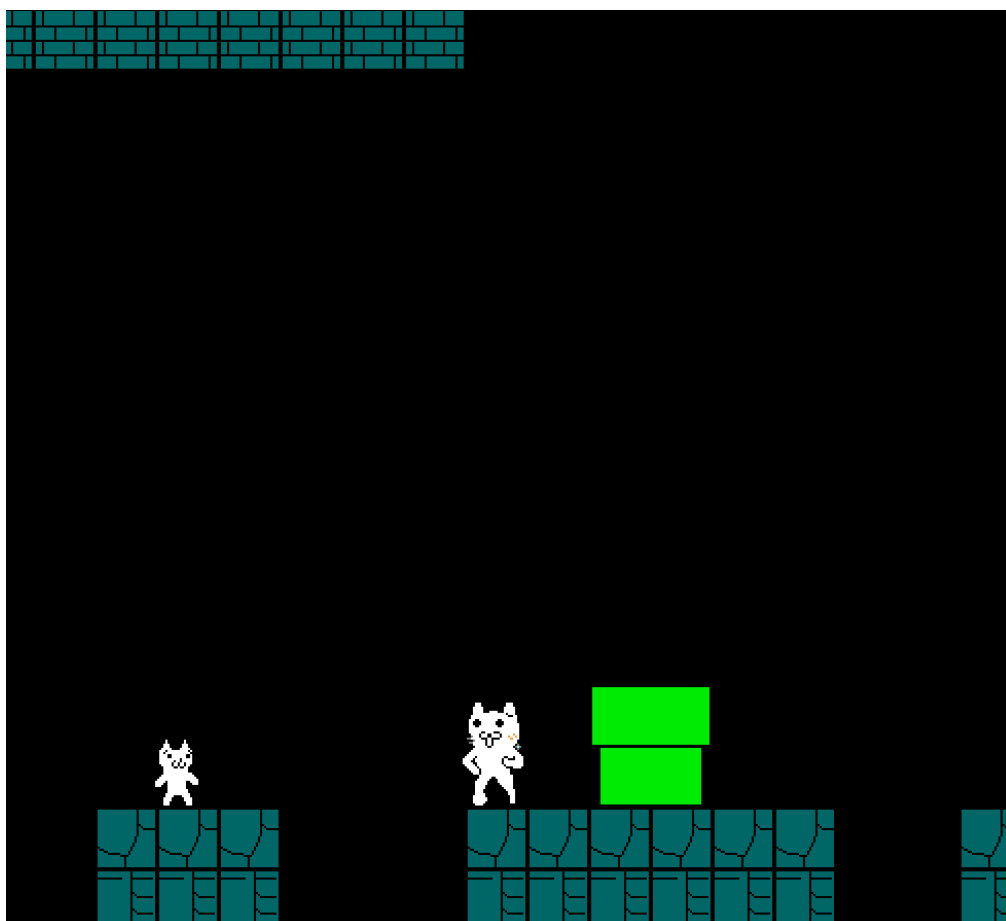
作弊模式開關:c 作弊模式可以無視所有怪物和陷阱，但掉下去一樣會死。

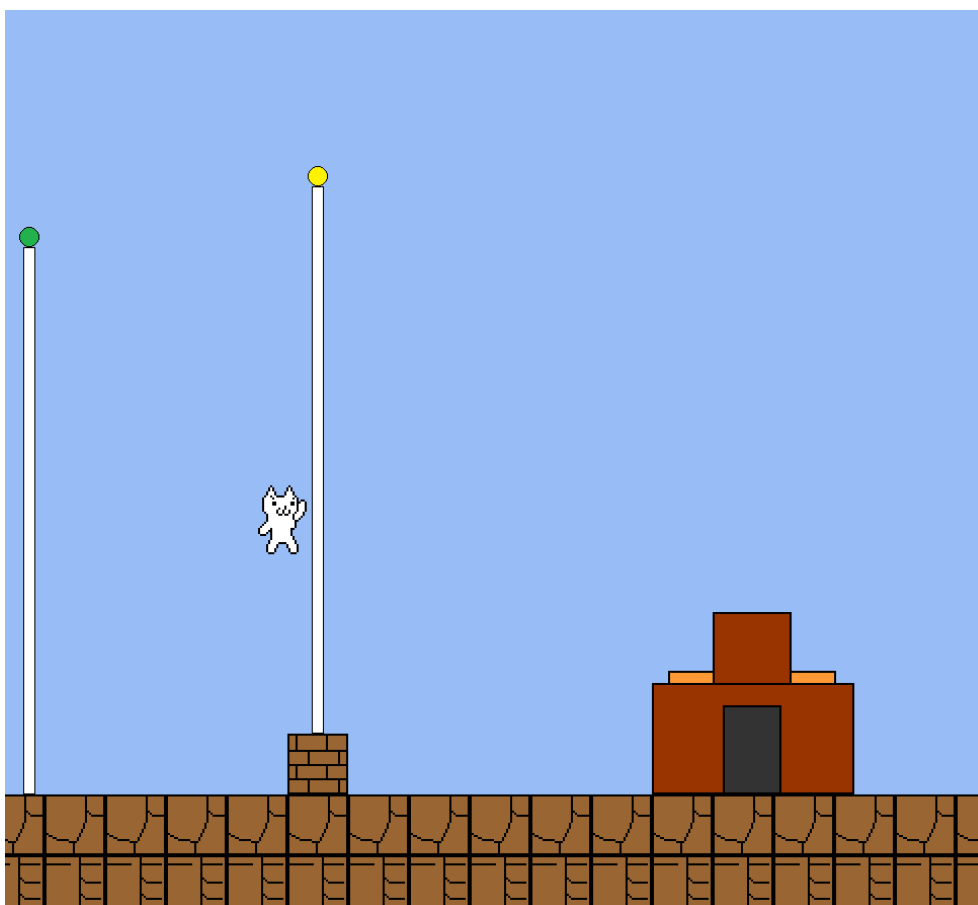
2. 遊戲圖形











3. 遊戲音效

以下陳列在遊戲中使用到的音效

音效類別	音效檔案
地上背景音樂	resources/audio/map_song/field.wav
地下背景音樂	resources/audio/map_song/dungeon.wav
跳躍	resources/audio/player_audio/jump.wav
金幣	resources/audio/interact_audio/coin.wav
吃蘑菇	resources/audio/player_audio/powerup.wav
進入水管	resources/audio/interact_audio/dokan.wav
死	resources/audio/player_audio/death.wav
通關	resources/audio/map_song/goal.wav

三、 程式設計

1. 程式架構

在這次實習，我們主要將架構分成遊戲運作和碰撞箱兩個部分，遊戲運作皆為類別 CGameStateRun 內添加的 function

- 遊戲運作
 - 事件處理(eventCtrl.cpp)
 - 地圖生成(mapSetting.cpp)
 - 碰撞處理(touch.cpp)
- 碰撞箱(colliders.h)
 - 碰撞箱(Class Collider)
 - 玩家(Class Player)(繼承Collider)
 - 方塊(Class Brick)(繼承Collider)
 - 敵人(Class Enemy)(繼承Collider)

2. 程式類別

類別名稱	標頭黨(.h)	原始碼(.cpp)	說明
CGameStateRun	52	1752	遊戲運作
Collider	13	20	碰撞箱
Player	56	227	實作玩家
Brick	19	12	實作方塊
Enemy	18	15	實作敵人
Environment	6	0	實作背景
總行數	164	2026	

3. 程式技術

(一)碰撞箱

我們參考了Unity的Component(元件)架構，來寫每種碰撞箱類別，用各種 boolean和int來實作Component，再用碰撞處理的function分別檢查每個碰撞箱是否擁有特定Component，而做出對應的處理。

比如說我想要某個方塊一開始是隱形的，在玩家的頭碰到方塊的底部時方塊會出現，那我就要在這個方塊添加"隱形"元件；在碰撞處理的function中，在"玩家的頭碰到方塊底部"的條件下 判斷這個方塊有沒有"隱形"而且玩家的縱向"速度"為負，如果判斷成立，這個方塊就會換成方塊圖片並讓它的"隱形"為false。

(二)物理相關

每個碰撞箱都有橫向和縱向的"速度"數值，如果物件有鋼體元件，就會每幀為它增加縱向的速度，再來就只要每幀根據速度設定座標即可。

四、 結語

1. 問題及解決方法

大部分的問題都圍繞在各種物件之間的碰撞。

首先是速度太快導致穿透地板的問題，玩家採方塊的碰撞判定是 檢查玩家的腳有沒有在"方塊頂部到方塊中間"的區間中，這個區間是30，因為有加速度的關係，玩家的縱向速度是可以超過30的，這導致玩家的腳直接略過判定區間，跑到方塊下半部，就穿透地板了。解決方法是 如果速度大於29，就會預測玩家下一幀的落點，如果超過地板判定區間，就會瞬間移動到地板上。當然還有各種比較小的穿透問題，我們也是東拼西湊才把問題解決。

再來是lag問題，由於方塊數量很多，每個怪物每幀都要和每個方塊做判定，只要怪物一多就會開始變慢。所以我們利用了玩家不能往回走的機制，刪除了後面的所有方塊與怪物；還有原本我們是一開始就把所有方塊生出來，後來改成玩家走到地圖中間時才生成下半部分的地圖，才勉強解決問題。

接下來是碰撞問題，

2. 時間表

- 韓正勤(110590008)

週	時間	說明
1	None	說明
2	2小時	練習
3	3小時	了解框架 構想程式寫法
4	4小時	腳色移動
5	7小時	腳色跳躍 與方塊碰撞
6	9小時	修復碰撞和跳躍
7	3小時	Demo1
8	8小時	修復碰撞 嘗試作怪物
9	5小時	重構程式碼
10	5小時	重構程式碼
11	12小時	重構完成並繼續做怪物和陷阱

12	8.5小時	Demo2(完成第一關)
13	6小時	修復第一關bug lag等問題
14	6小時	把第一關沒做完的部分做完
15	3小時	做第二關怪物
16	8小時	第二關陷阱怪物並修復bug
17	7.5小時	Demo3(完成第二關)
18	None	還沒到
總計	97小時	

- 張塔信(110590039)

週	時間	說明
1	介紹	介紹
2	3 小時	Gather and Create game resources.
3	6 小時	Create a factory
4	10 小時	Create block collision
5	8 小時	Fix collision bugs and help group mate fix camera movement bug
6	6 小時	Build a map1 and draw an image for the game resource.
7	3 小時	Create animation
8	4 小時	Add audio in game
9	14 小時	Create a player dead screen , more player animations and create a

		checkpoint.
10	10 小時	Fix the animation bugs.
11	25 小時	Code refactoring
12	6 小時	Fix map1 bugs
13	12 小時	Create map2
14	12 小時	Fix map2 and game audio bugs
15	11 小時	Add more traps on map2 and fix enemy collision bugs
16	1 小時	Change enemy size and a sound effect.
17	3小時	Fix flag bug
18	None	
總計	134 小時	

3. 貢獻比例

大概就是各占五十趴，我們每周都有分配工作，有人沒做完會幫對方做

4. 自我檢核表

	項目	是否完成	說明為何不能完成
1	解決Memory Leak 的問題	是	
2	自訂遊戲 Icon	是	
3	About 畫面	是	
4	遊戲中說明如何操作	否	本遊戲沒有任何界面 為了遊戲畫面的整潔 我們把操作放在About中
5	發布(Release)版本	是	
6	承上項，該版本可以正常運作	是	
7	報告字型、點數、對齊、行	是	

	距、頁碼等格式正確		
--	-----------	--	--

5. 收穫

我學到了如何有效管理不同類別之間的互動。在這過程中，我進行了一次中間重構，從中我獲得了一個重要的教訓：在進行大型專案時，先構想好所有相關事項非常重要，以免在開始實作後逐漸變得雜亂無章。同時，我也學到了如何與他人合作，共同完成一個專案。

6. 心得或感想

我對遊戲開發很感興趣，再加上有助教寫好的框架可以用，讓我開發起來很有動力，所以這堂課上起來特別開心。

7. 對於本課程的建議、想說的話

感謝助教