

# 第9組

# Sokoban—Stealer

# 期末報告

小組成員：

00771053 李佳勳

0076D053 沈彥昭

0053A018 吳永璿

## Sokoban—Stealer

使用Java.swing實作經典的2D遊戲Sokoban, 並額外加入警衛和穿牆能力等元素。

『《倉庫番》是一款經典電子遊戲, 1982年由日本Thinking Rabbit公司首次發行。之後其他遊戲開發者爭相製作仿製或衍生作品。致使倉庫番成為此類遊戲的代名詞。遊戲要求玩家在二維地圖上把箱子推到指定地點, 當中牽涉到大量的空間邏輯推理。』

—維基百科

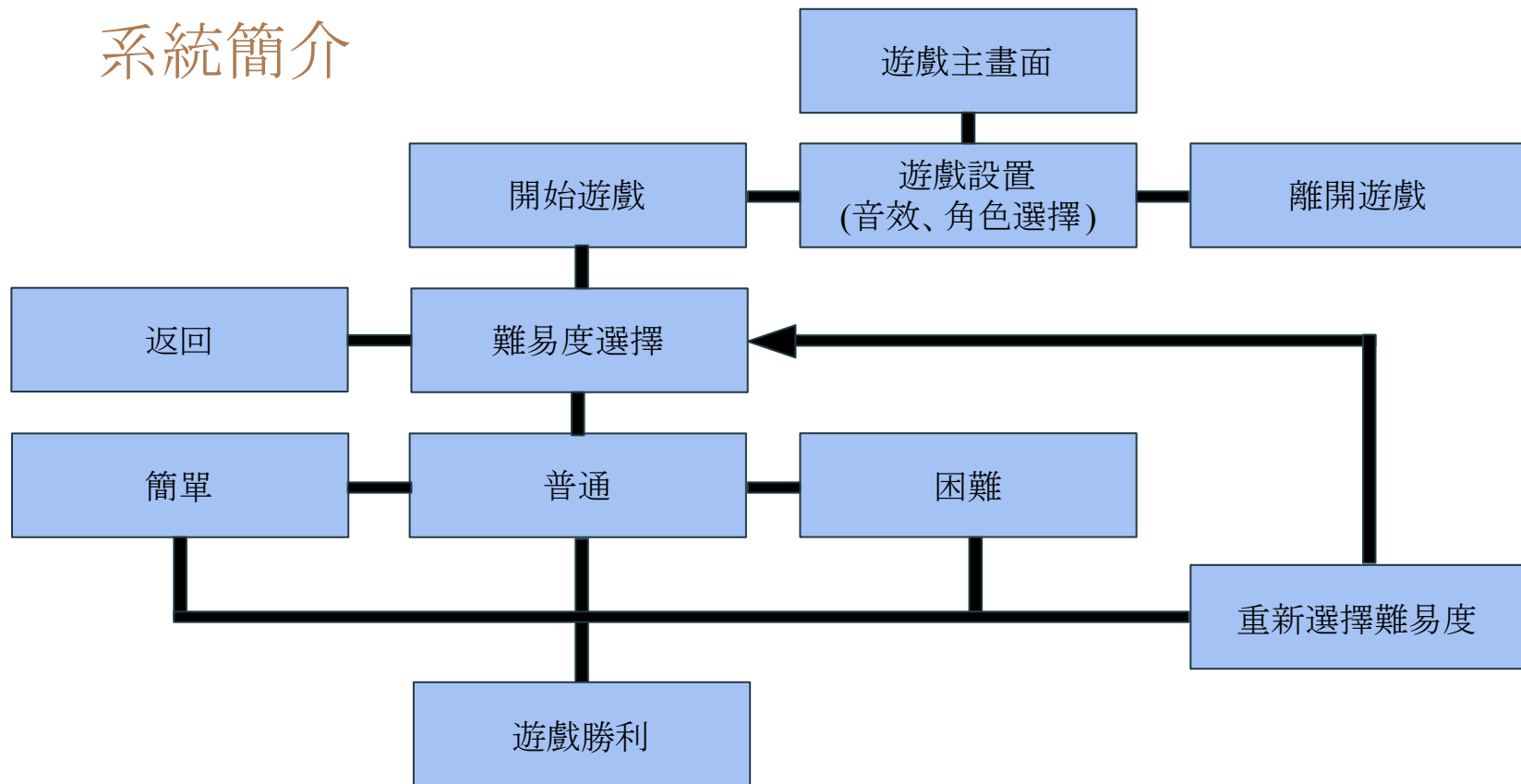
## 遊戲大綱：

遊戲難度有分簡單，普通，困難三個等級，玩家可以自行選擇並遊玩，每關有三次機會。

每種難度皆有數個關卡，玩家需扮演小偷，以「推」的方式推動箱子。在沒有阻礙物(如牆壁等的阻礙物)的情況下，可以向上、下、左、右的方向移動。將箱子移動到指定地點，並達到指定數量即可過關。

推動過程需注意並躲避警衛，被發現及抓捕則出局。必要時可以使用有限制次數的穿牆能力逃離。若達成一定個數的貨物，可以獲得槍，用來對付警衛。另外遊戲中會有可以使用傳送的能力來躲避警衛，依玩家如何使用甚至可以一舉逆轉危險的情況。

# 系統簡介



## 分工

美術設計、穿牆技能、音效選擇:沈彥昭

地圖配置、警衛製作、人物類別:李佳勳

主選單、地圖類別、槍、傳送門:吳永璿

Code :全員

## 時程

5/4-5/10 學習網路上提供的教學

5/11-5/17 實作地圖類別

5/18-5/24 各自完成類別, 加入音效

5/25-5/31 討論、修改

6/1-6/7 內容整合, 測試

# 使用到的API

`java.util.timer`

警衛類別, 穿牆, 子彈, 畫面自動刷新

`java.swing`

遊戲畫面呈現

## 遭遇的問題

## 解決方法

git hub使用	慢慢摸
eclipse專案圖片路徑跟VS code不同	加if else個別處理
傳送門引起人物蓋住貨物神奇bug	改成有貨物蓋住時不可傳送
子彈可以穿過地圖	子彈撞牆就free掉
穿牆穿出地圖	增加hard wall
人物正面碰到警衛後僅把visible設false, 遊戲未關閉	遊戲結束時把frame dispose
關卡complete會當掉	關卡complete把frame dispose

# 版本開發歷史

v1.0.0 - 基礎遊戲玩成

v1.0.1 - 修正貨物可推至hardWall及穿牆可多次使用的bug

v1.1.0 - 警衛版一更改 增加警衛class 在board裡增加checkBagCollisionforPolice method

v1.1.1 - 圖片更改;圖片封裝優先度更改, 警衛的圖片會顯示在較上層

v1.2.0 - 新增地圖, 更改路徑。

v1.2.1 - 將警衛改為不可穿牆。為debug將穿牆功能暫時改為按一次即可永久穿牆, 再按一次恢復, 交替運作

v1.2.2 - 配合eclipse修改圖片載入路徑

v1.3.0 - 人物會依據移動方向改變圖片方向;將穿牆功能改為X鍵、傳送點功能改為Z鍵, 更改package

v1.3.1 - 警衛碰到子彈就消失, 人物正面碰到警衛遊戲結束

v1.3.2 - 修改警衛消失後的bug,修改成警衛子彈不同調

v1.4.0 - 子彈設定成無法連射, 難度決定警衛跑速, 增加主選單

v1.4.1 - 新增圖片;精簡部分程式碼;新增一張地圖(地獄);警衛改成每走兩步換行進方向, 減少混亂感

v1.4.2 - 現在關卡完成後遊戲會正確關閉了

v1.4.3 - 對齊貨物至格子中間, 削減部分程式碼, 新增一首BGM(尚未能撥放), 暫時新增二號玩家

v1.4.4 - 警衛2.0

v1.5.0 - 最終版



# 製作時程

## 預定:

5/4-5/10 學習網路上提供的教學

5/11-5/17 實作地圖類別

5/18-5/24 各自完成類別, 加入音效

5/25-5/31 討論、修改

6/1-6/7 內容整合, 測試

## 實際:

5/4-5/10 開發遊戲主程式

5/11-5/17 實作警衛及傳送門等特殊物件

5/18-5/24 開發新地圖

5/25-5/31 開發未完成之功能

6/1-6/7 最終測試、整合

# 製作此專案之心得

吳：一個人大概會寫很久，有隊友一起寫輕鬆很多。溝通很重要。

沈：學習到不少java特有的語法，但也遇到不少困難。雖然製作出來的成果跟當初預期的有些不同，但是最後也是做出了一個不錯的遊戲，有不少收穫。

李：第一次做程式小組分工，深刻了解變數名稱要訂的好理解一點，最好加一些註解，若沒有多做這些步驟，其他人可能要多花一倍的時間去理解到底在做啥。知道程式可讀性在小組分工重要性。