

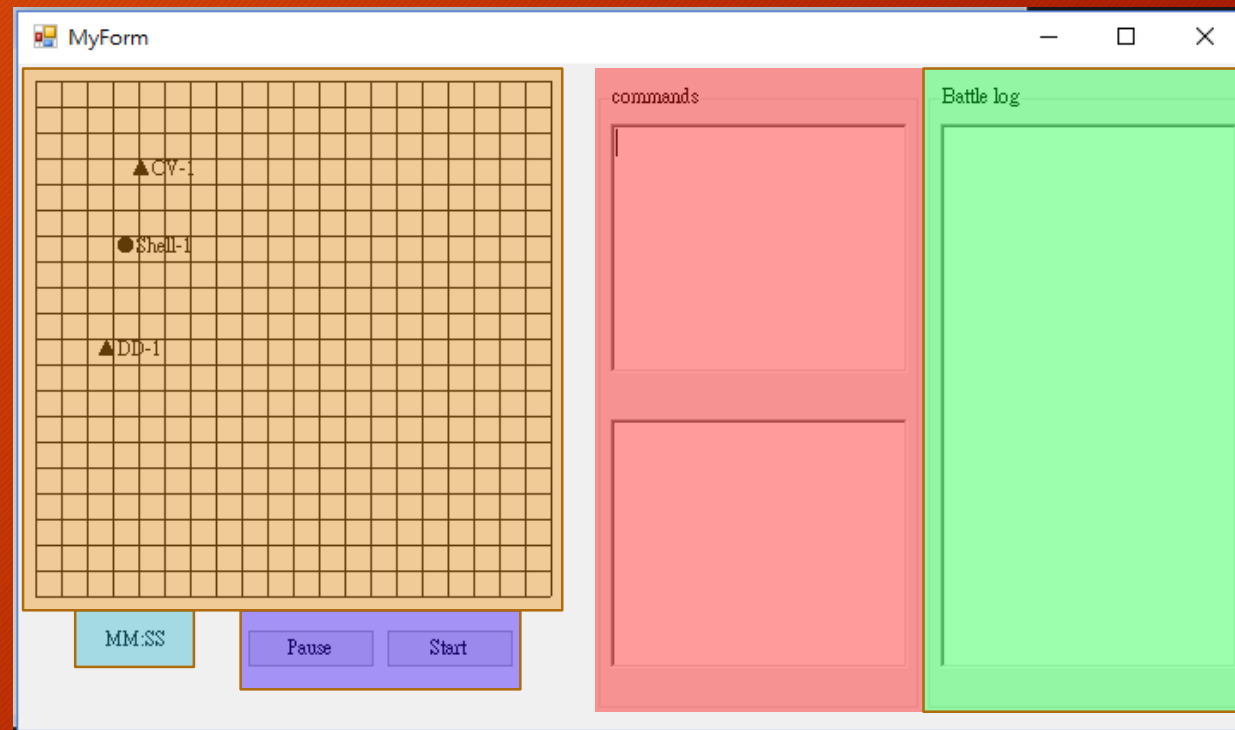
Project 3 即時戰略遊戲

OOP-Lab

概要

2

- 做出一個海戰模擬器，觀測雙方的動態並顯示出來。
- Input:
 - 戰鬥指令輸入區(紅色)
 - A艦隊(上)
 - B艦隊(下)
 - 開始/暫停鍵(紫色)
- Output:
 - 海戰動態(橘色)
 - Battle Log(綠色)
 - 進行時間
(MM:SS，藍色，顯示遊戲內時間)



規則

3

- 地圖大小 20*20 平方單位。
 - (0, 0)在左上，往右下遞增。
- 遊戲內時間為15倍速 (現實1秒 -> 遊戲15秒)。
- 地圖動態每秒(現實時間)更新一次。
- **Battle Log** 時間戳記使用遊戲時間。
- 任何時間都可按暫停鍵輸入指令。
輸入指定的時間為現實時間，不需準確到以遊戲時間紀錄。
- 船艦主要可進行移動、攻擊、防禦。
- 因為是上帝視角，所以雙方艦隊的動態都會呈現。
- 沒有遊戲結束的條件，所以沒有終點也沒有輸贏。
就算雙方都沒有船艦了，遊戲也不算結束，仍可再新增船艦繼續。

艦種

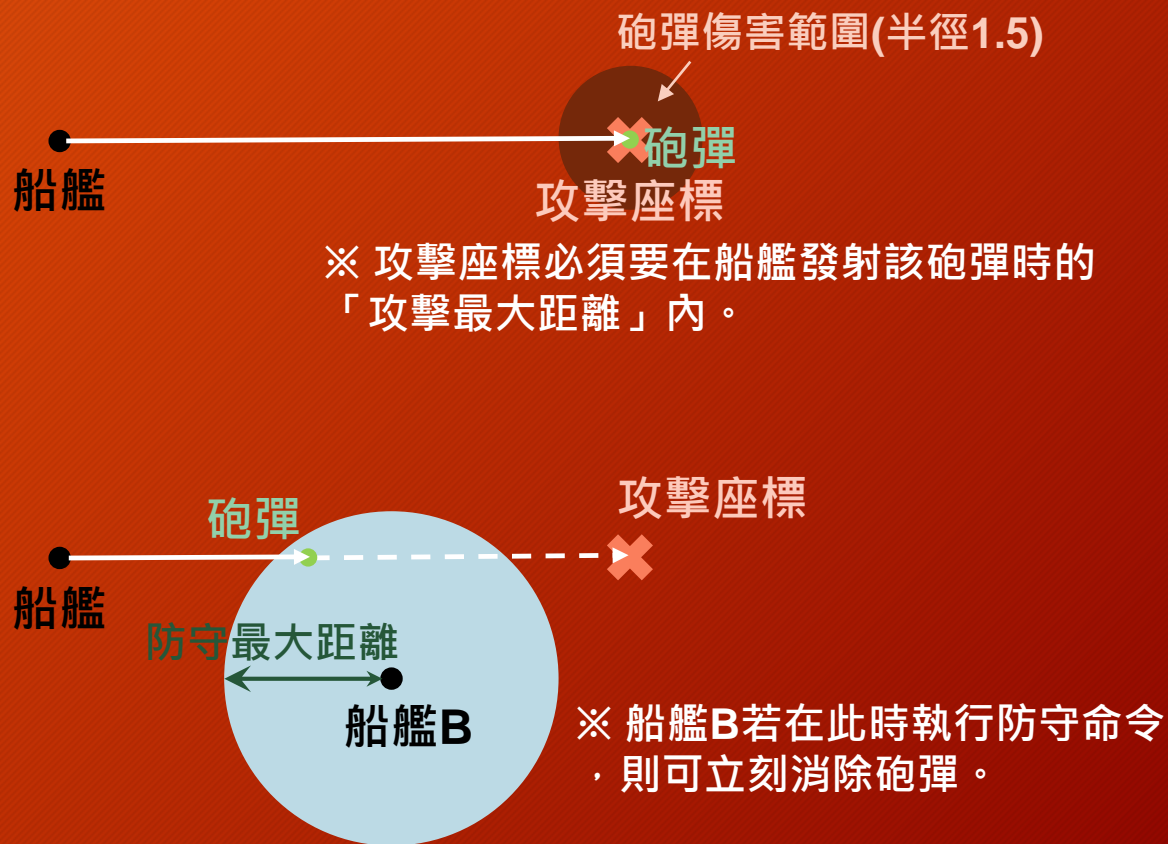
4

艦種名稱	HP	最高航速 (單位長度/分鐘)	攻擊最大距離	攻擊CD (秒)	防守最大距離	防守CD (秒)	武器傷害	砲彈飛行速度 (單位長度/分鐘)
CV	5	1	25	60	5	15	3	4
BB	4	1	20	30	10	30	3	2
CG	3	2	15	30	15	30	2	3
DD	2	3	10	15	20	60	1	3

- ※ 同個時間，不同船艦可以出現在同個位置(沒有碰撞機制)。
- ※ 砲彈造成的傷害範圍為半徑1.5單位長度，到達指定位置後爆炸(才會產生傷害)，但若有船艦在該砲彈行進路徑上使用防守，則可消滅該砲彈。
- ※ 存在同隊傷害。
- ※ 本表使用的時間為遊戲時間。

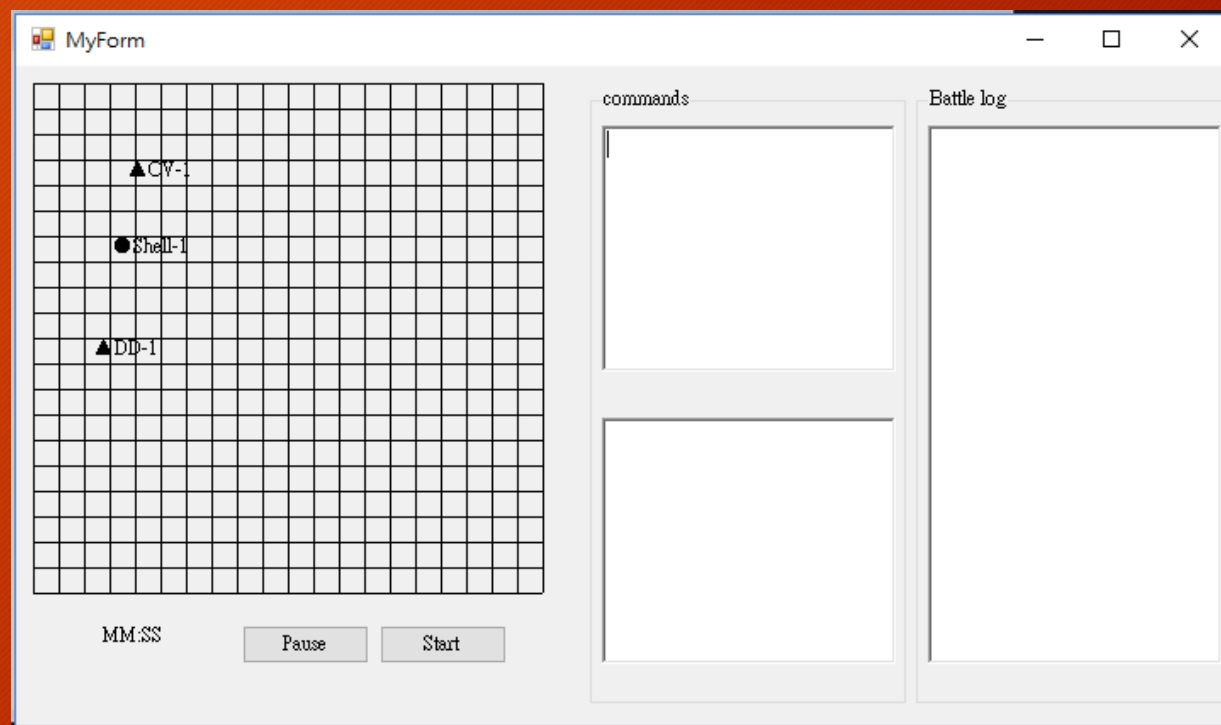
艦種

5



基本指令(系統指令)

- 開始計時
 - 按下 Start Button後，不可輸入戰鬥指令。
- 暫停計時
 - 按下 Pause Button後，可輸入戰鬥指令。
- 時間顯示
 - 顯示遊戲內時間。Ex: 04:30(MM:SS)
 - 現實中的1秒 = 遊戲中的15秒
- 【補充】
 - 所有戰鬥指令只能在暫停時輸入。
 - 開始時才會繼續執行戰鬥指令。
 - 戰鬥指令有順序性，先執行A艦隊再執行B艦隊。



※必須要做清空輸入區的按鈕或指令

基本指令(戰鬥指令)

- **SET [Vessel_Name] [Type] [2DCoordinate]**

- 新增船艦於某座標：設定新船艦名字[Name]、艦種[Type]、座標[2DCoordinate]。
- 場上同隊的船艦名字不能重複，但若A被擊沉了，就可再新增一艘A。
- Type必須出現在艦種裡。
- [2DCoordinate]的範圍在(0.0~20.0, 0.0~20.0)。
- Ex: SET CV-1 CV (0.0, 12.0)

- **FIRE [Vessel_Name] [2DCoordinate]**

- 發射砲彈攻擊某座標：攻擊船艦名字[Vessel_Name]、攻擊座標(中心點)[2DCoordinate]。
- 攻擊船艦必須在場上且存在於該隊。
- [2DCoordinate]的範圍在攻擊最大距離內。
- 自動對該砲彈命名，砲彈名字: Shell_(team+id)，其id為發射順序。
(ex. Shell_A1為A艦隊第1個發射的砲彈)
- 攻擊後就進入CD時間
- Ex: FIRE CV-1 (0.0, 10.0) //Shell_A1 出現，CV-1進入攻擊CD(冷卻)時間

基本指令(戰鬥指令)

- **DEFENSE [Vessel_Name] [Shell_Name]**

- 防守某砲彈：防守船艦名字[Vessel_Name]、預計防守的砲彈名字[Shell_Name]。
- 防守船艦必須在場上且存在於該隊。
- Shell_Name必須存在於場上。
- 若無法攻擊到該砲彈則視為未命中。
- 不論命中與否，執行後進入防守CD(冷卻)時間。
- Ex: DEFENSE CV-1 Shell_A1 //CV-1進入防守CD(冷卻)時間

- **TAG [Vessel_Name] [New_Name]**

- 重新命名船艦：船艦舊名字[Vessel_Name]、船艦新名字[New_Name]。
- 船艦必須在場上且存在於該隊。
- 新名字不能與場上同隊船艦名字相同。//舊名跟新名不能一樣
- Ex: TAG CV-1 ALPHA

基本指令(戰鬥指令)

9

- **MOVE [Vessel_Name] [Speed] [Angle]**

- 船艦移動：船艦[Vessel_Name]向角度[Angle]以[Speed]的速度前進。
- 船艦必須在場上且存在於該隊。
- 速度絕對值不得超過最高航速。
- Angle為0~359度, 0度為正右方，逆時鐘方向旋轉。
- 對同船艦下達的移動指令，只會覆蓋前一條，不會累加。
- 當船艦移動到邊界時，則停止移動。
- Ex: MOVE CV-1 0.5 90

Battle Log (戰鬥指令對應輸出)

10

- **SET [Vessel_Name] [Type] [2DCoordinate]**
 - [mm:ss] [team] SET [Vessel_Name] with [Type] at [2DCoordinate] -> Success
 - [mm:ss] [team] SET [Vessel_Name] with [Type] at [2DCoordinate] -> Fail
- **FIRE [Vessel_Name] [2DCoordinate]**
 - [mm:ss] [team] [Vessel_Name] FIRE to [2DCoordinate] -> [Shell_Name]
 - [mm:ss] [team] [Vessel_Name] FIRE to [2DCoordinate] -> Fail
- **DEFENSE [Vessel_Name] [Shell_Name]**
 - [mm:ss] [Vessel_Name] DEFENSE [Shell_Name] -> Hit
 - [mm:ss] [Vessel_Name] DEFENSE [Shell_Name] -> Fail
- **TAG [Vessel_Name] [New_Name]**
 - [mm:ss] [team] RENAME [Vessel_Name] to [New_Name] -> Success
 - [mm:ss] [team] RENAME [Vessel_Name] to [New_Name] -> Fail
- **MOVE [Vessel_Name] [Speed] [Angle]**
 - [mm:ss] [team] [Vessel_NameA] MOVE to [Angle] as [Speed] -> Success
 - [mm:ss] [team] [Vessel_NameA] MOVE to [Angle] as [Speed] -> Fail

Battle log

```
[00:00] TeamA SET CV-1 with CV at (2,2) -> Success  
[00:00] TeamA SET CV-1 with CV at (2,3)->Fail
```

```
[00:10] TeamA CV-1 Fire to (3,9) -> Shell_A1  
[00:11] TeamA CV-1 Fire to (3,9) -> Fail
```

```
[00:10] CV-1 DEFENSE Shell_A1 -> Hit  
[00:11] CV-1 DEFENSE Shell_A1 -> Fail
```

```
[00:10] TeamA RENAME CV-1 to Alpha -> Success  
[00:11] TeamA RENAME CV-2 to Alpha -> Fail
```

```
[00:10] TeamA CV-1 MOVE to 90 as 0.5 -> Success  
[00:11] TeamA CV-1 MOVE to 90 as 2 -> Fail
```


Battle Log (盤面變化對應輸出)

11

- 除了顯示戰鬥指令，也需顯示 System(盤面變化)。
- 輸出時機
 - 砲彈落地後沒有擊中
 - [mm:ss] [Shell_Name] arrived [2DCoordinate] -> miss
 - 砲彈落地後有擊中
 - [mm:ss] [Shell_Name] arrived [2DCoordinate] -> hit {[team]+[Vessel_Name], ...}
 - 船隻被擊沉
 - [mm:ss] [team] [Vessel_Name] destroyed

```
[00:10] Shell_A1 arrived (3,9) -> miss  
[00:11] Shell_A2 arrived (3,9) -> hit {TeamACV-1, TeamBDD-1}  
[00:11] TeamA CV-1 destroyed  
[00:11] TeamB DD-1 destroyed
```

- 【備註】事件同時發生時的輸出順序為 A艦隊->B艦隊->System

必須實作項目(70%)

12

- 控制介面

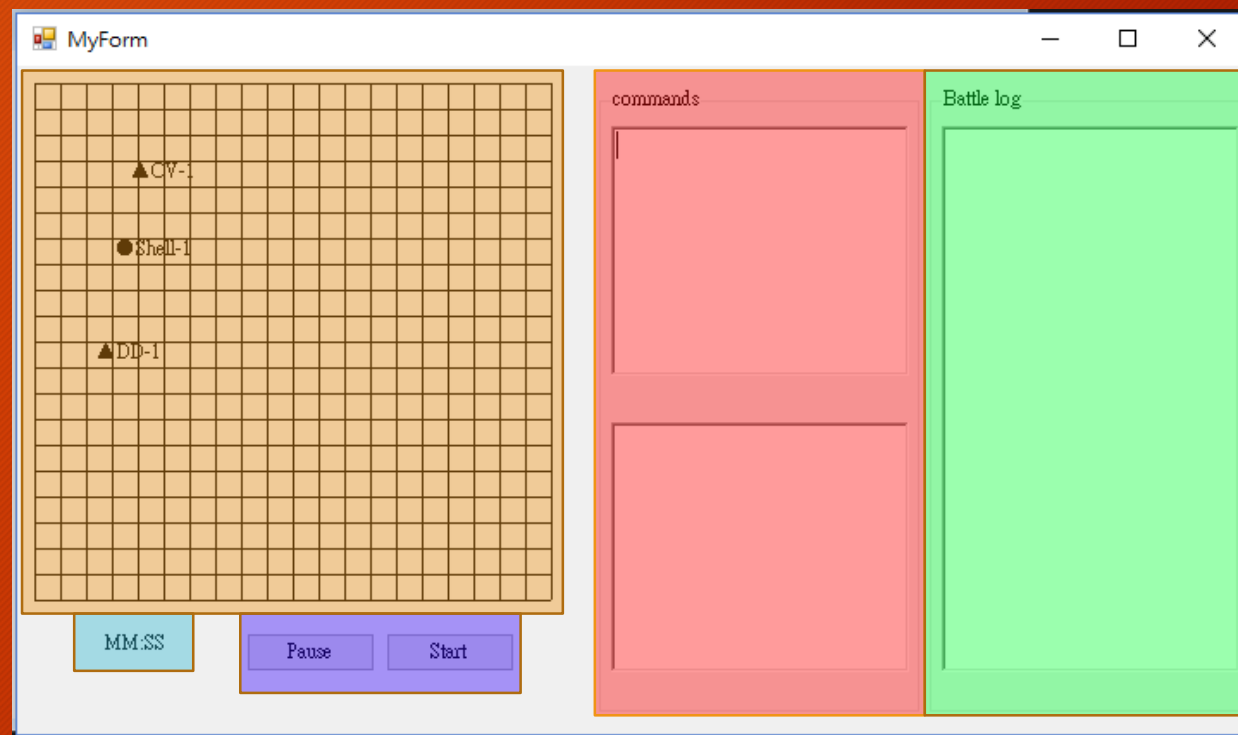
- 開始鍵、暫停鍵、時間顯示
- 海戰動態圖
 - 能明顯區分各艦隊的船隻與火炮。(可使用顏色或圖示區分)
- 各艦隊的戰鬥指令輸入區
- Battle Log 顯示區

- 程式

- 完成所有基本指令。
- 隊伍、船艦、砲彈皆須使用class實做。
- 所有艦種須由一個VesselObject繼承。
- 所有砲彈射出後必須被創立一個物件。
- 計時器與Battle Log可正確運行。

- 說明書(pdf)

- 介面說明、額外增加的功能介紹與操作說明(EX: 新增的船艦、砲彈、命令)。



選擇實作項目

- 更多模式及呈現方式，例如：3D地圖、提供能遊玩的模式(主觀分數，50%)。
- 增加其他艦種(5%)
- 增加多樣武器選擇(5%)
 - Ex: 增加導彈(可直接瞄準船艦)...等。//可能需要調整指令
- 增加船艦互相碰撞
 - 碰撞後停止(10%)
 - 船艦交會後停下。
 - 碰撞傷害(10%)
 - 船艦交會後傷害，但不一定要停下。
- 天氣水文系統(10%)
 - 需要用圖像呈現Ex: 漩渦、颱風、暴風雨...等影響遊戲因素。

選擇實作項目

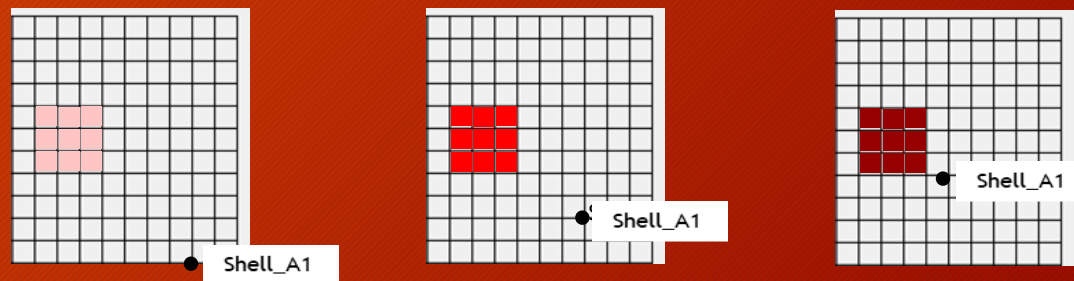
14

- 新增地形(圖像化)

- 船艦無法穿越之地形(5%)
- 多種海底、陸上地形(5%)
 - Ex: 海底、陸上地形有高低之分，平地砲彈可穿過，高山砲彈則無法，或是暗礁事件等等。
- 進入遊戲前可設定(5%)
- 遊戲中可以調整(5%)

- 新增區域警示

- 文字顯示(5%)
- 圖像化顯示(15%)
 - 提示: 用顏色變化表示目標區域危險程度
 - 例如：砲彈越靠近目標區域越深 (EX: 剩15s -> #FF0000)。



評分方式

15

- 必須實作項目全數完成後，才會評測選擇實作項目。
- 主觀分數沒有負分。
- 製作專案時請必須考量評測方便性，若難以辨別功能是否完成則視為未完成。
- 任何情況下出現 runtime error -> -20分
 - Error handling 請做好。
- 圖形介面可以不以 Windows Form 實作，但要用C++(Ex:Qt、Unreal、GLUT)。
- 最高分100。
- 本次作業不提供範例程式。

- VS2017 CLR 產生方法(英文，附圖) <https://ppt.cc/fGcaux>
- 網格圖怎麼拉(超級陽春作法) <https://imgur.com/a/mRIHYnM>