

碰撞偵測 & 處理 ★

責任鏈模式——碰撞偵測 & 處理

難度：★☆☆☆☆

1. 你需要開發一個簡單的 DEMO 程式，程式的運作如下：

1. 程式開始時首先會準備好世界 (World)：

1. 這個世界是一維的 (長度為 30)。世界中存在著許多生命 (Sprite)，初始有 10 個生命，每個生命都會被隨機賦予一個初始的座標 (數值範圍 0~29；0 代表世界的最左側，29 代表世界的最右側) 代表他的初始位置。

2. 初版的世界存在以下三種生命：

1. Hero (在世界中顯示為 H)，Hero 擁有生命值 (HP)，生命值的初始值為 30。
如果 $HP \leq 0$ 時，Hero 死亡，並且會從世界中被移除。

2. Water (在世界中顯示為 W)

3. Fire (在世界中顯示為 F)

2. 程式開始後，會請使用者輸入兩個數字 (以空白隔開)，第一個數字為 $x1x1$ ，第二個數字為 $x2x2$ 。輸入這兩個數字代表使用者想要將位於位置 $x1x1$ 的生命移動到位置 $x2x2$ 。

3. 如果處於位置 $x1x1$ 的生命 ($c1c1$) 想要移動到 $x2x2$ ，而位置 $x2x2$ 上存在著另一個生命 ($c2c2$) 的話，此時就會觸發 $c1c1$ 和 $c2c2$ 之間的碰撞 (Collision)，不同生命種類組合的會有不同的碰撞效果：

1. **Water 與 Fire 的碰撞效果：**

1. Water 從世界中被移除。
2. Fire 從世界中被移除。

2. **Water 與 Water 的碰撞效果：** 移動失敗。

3. **Fire 與 Fire 的碰撞效果：** 移動失敗。

4. **Hero 與 Fire 的碰撞效果：**

1. Hero 生命值減少 10 滴。
2. Fire 從世界中被移除。
3. 如果 $c1c1$ 為 Hero， $c1c1$ 移動成功。

5. **Hero 與 Water 的碰撞效果：**

1. Hero 生命值增加 10 滴。
2. Water 從世界中被移除。
3. 如果 $c1c1$ 為 Hero， $c1c1$ 移動成功。

6. **Hero 與 Hero 的碰撞效果：** 移動失敗。

4. 處理完碰撞效果之後，移動結束，回到步驟 1-b，無限循環。

設計需求

你的程式碼必須在「擴充新的生命種類 → 擴充新的碰撞效果」的部分遵守著開閉原則 (Open-Closed Principle)。意思就是每當要擴充新的生命種類，因此要擴充新的碰撞效果時，開發者不必進到既有領域模型的類別中作修改，只需要在外部透過依賴注入等手段擴充新的碰撞效果。

如果你把碰撞效果的處理撰寫在「世界」類別中的話，是不可能遵守 OCP 的。

小提示

將任「生命種類碰撞組合」視為是一個「輸入」，而你的程式要想辦法處理各種「生命種類碰撞組合」。