



Group Project

GALAXY WARS

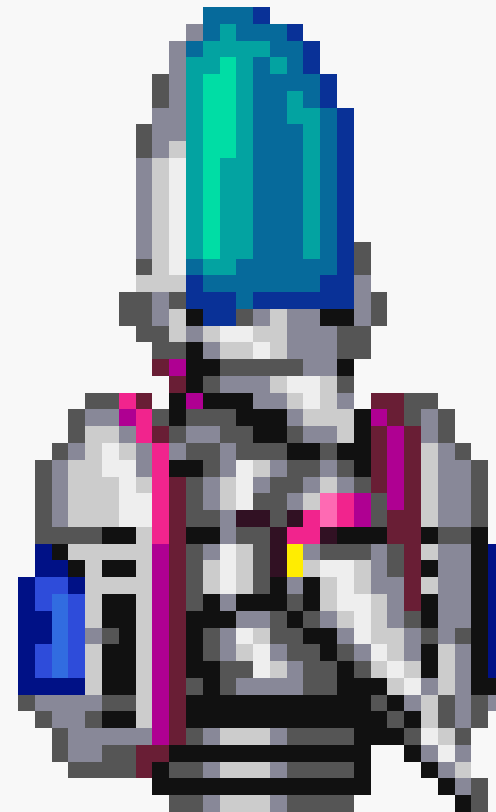
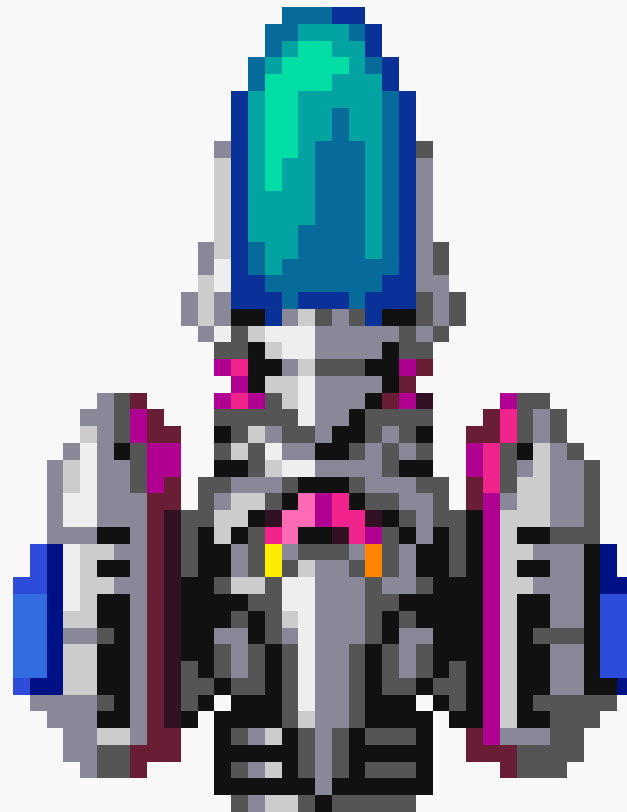
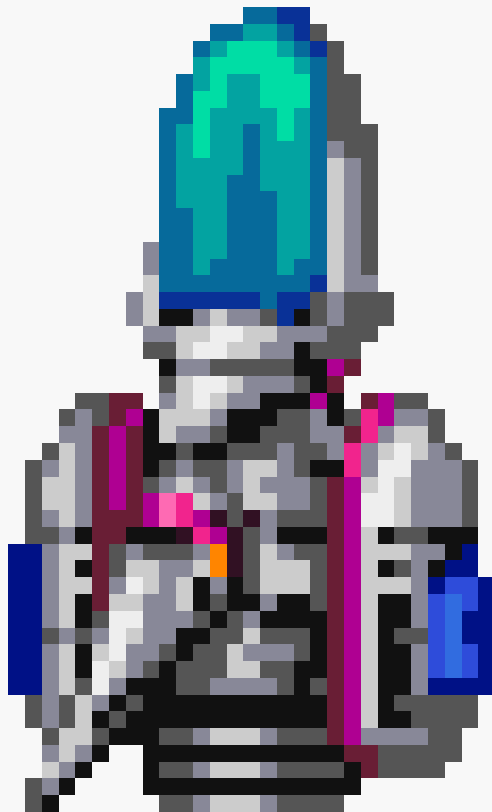
112101013 陳楷翔

112511275 陳柏佑



操作

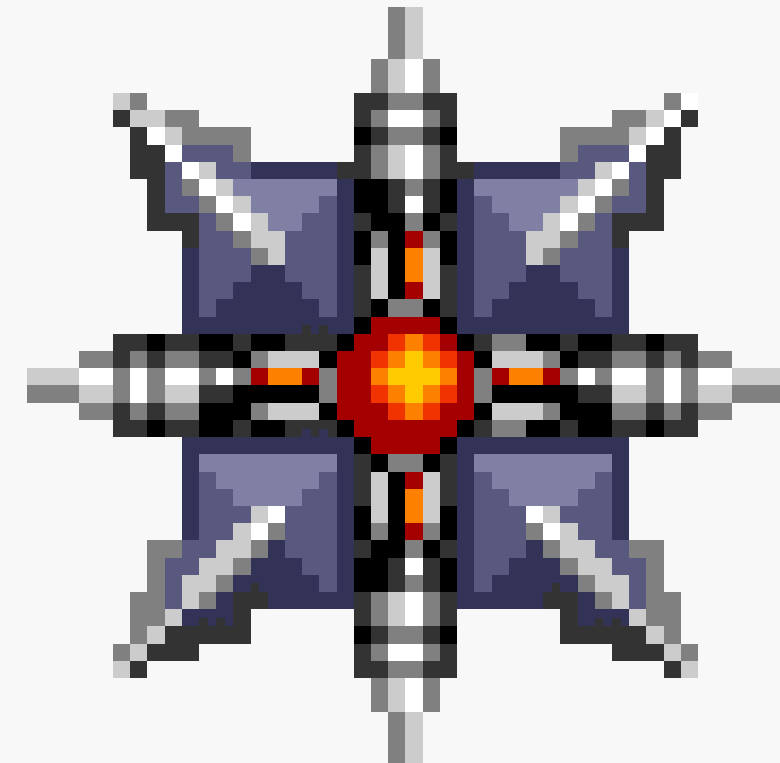
- 方向鍵：控制飛船方向
- 空白鍵：向前射擊
- Z 鍵：全範圍射擊



規則



- 擊落敵機：可提升玩家 shield 值以及 score 值
- shield 值：玩家受傷減少，歸零則遊戲結束，可消耗 30 點進行全範圍射擊
- score 值：累積至 1500 點時將召喚 Boss
- 脆弱狀態：約 1 秒，玩家受傷時觸發，期間不可發動攻擊，也不會受到傷害
- Boss：隨著血量減少加快移動速度，會發射散彈



物件分類：以「移動模式」為主要標準

- Player：移動方向可控，class內包含方向鍵感測與脆弱狀態表現函數
- Missile：移動方向固定，class內包含前進函數以及存在/消滅狀態參數
- Enemy：移動方向隨機，class內包含移動函數以及碰撞偵測函數

主程式：

- Utility：用於保存跨物件的功能，比如兩物件間距離偵測函數、字體繪製函數
- Game：控制程式畫面索引，決定當下要如何呈現遊戲，調用各種物件並組合各自的函數，每秒更新畫面30次，重複執行內部程式，直到畫面索引改變宣告遊戲中止

重要功能介紹：

- `set_enemy(x, y, angle, typ, spd, shield)`：呼叫此函數後會根據所提供的參數生成敵機，其中`spd`與`shield`分別代表，敵機速度與血量，為調整遊戲難度的主要函數
- `check_collision_with_player(self, player)`：呼叫函數後會先取得`self`物件的圖檔長寬，並用於計算碰撞半徑，接著利用`get_dis(self.x, self.y, player.x, player.y)`值是否小於碰撞半徑的平方來判定是否與玩家碰撞

Pandas 與 *matplotlib* 應用：記錄玩家遊戲數據

- 當敵機被破壞，紀錄各機種擊墜數，計算並存於 **kd list**
- 紀錄子彈總射擊量，以及對敵機之命中數，計算並存於 **mi list**
- 將 **kd list** 與 **mi list** 分別製成圓餅圖，清楚呈現擊墜比例以及命中率



Thank you

