**I. Introduction**

本程式為以C++語言所構成之單人文字冒險遊戲，運用物件打包、物件繼承、Virtual function等特性進行實作；使用者在遊戲中扮演地下城冒險者，透過鍵盤輸入來控制遊戲進行，本程式除了**達成所有要求之基本功能**，為進一步增加遊戲趣味性再加入了額外功能，報告中將詳細介紹。

**II. Implementation**

**1. 程式執行概要**

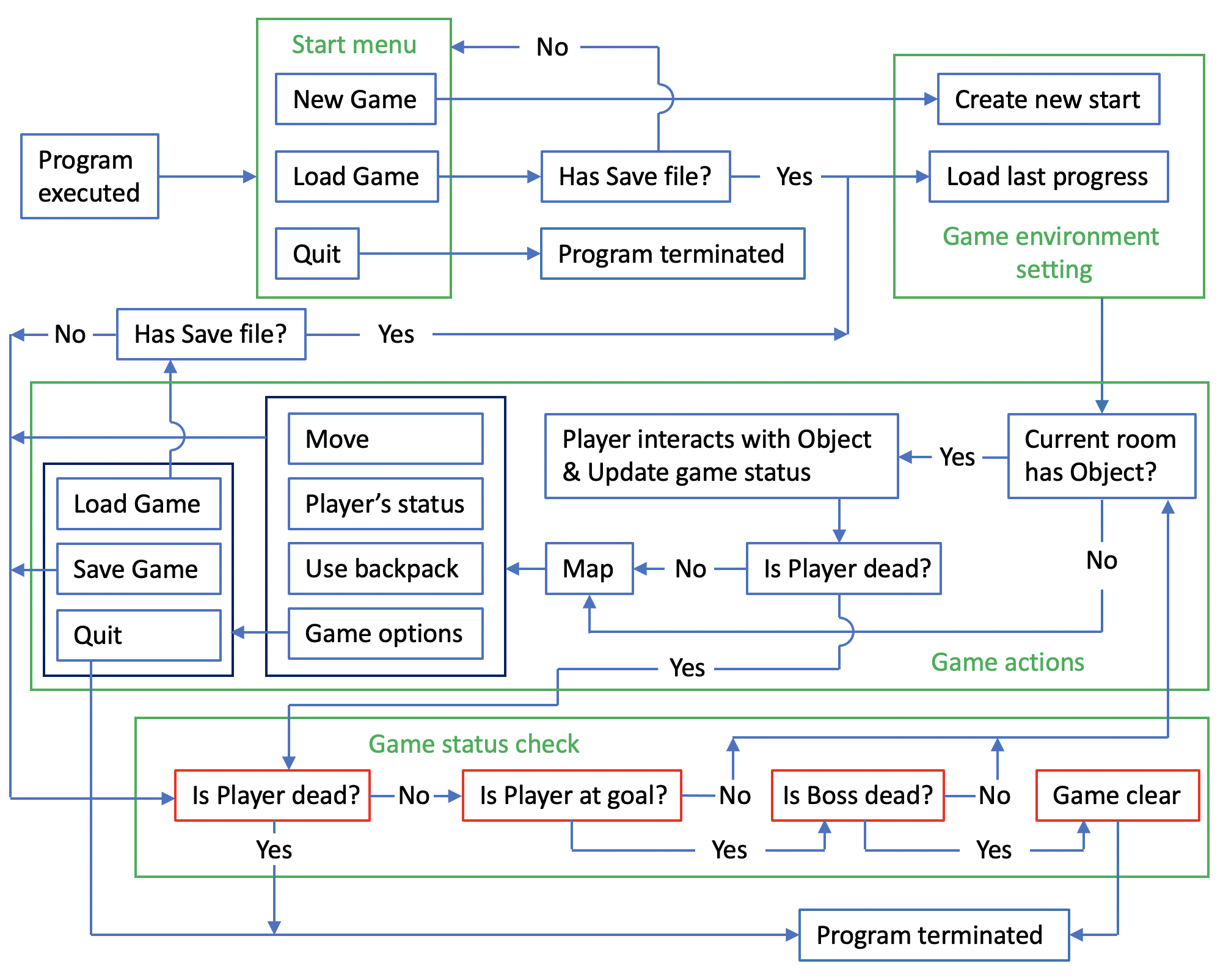


圖1: 程式運作流程圖

圖1為程式運作流程圖，主要包含4項功能：**開始選單(Start menu)**、**遊戲環境參數設定(Game environment setting)**、**遊戲動作(Game actions)**、**遊戲狀態檢查(Game status check)**，下面分別對以上功能介紹：

**(a) 開始選單 & 遊戲環境參數設定：**

開始選單為程式執行後讓使用者選擇哪種遊戲環境參數設定；選擇新遊戲(New Game)的情況下，會指示程式以預設數值來設定，大多數為固定數值，一些物件參數使用了隨機函數來生成；選擇讀檔(Load Game)的情況，會先判別是否有存檔資料(Has save file?)，有則用存檔數值來設定遊戲環境，沒有則讓使用者重新在開始選單做選擇；選擇離開(Quit)則會讓程式直接終止。

**(b) 遊戲動作：**

此功能主導了大部分遊戲的進行；首先會檢查角色(Player)目前房間是否含有物件(object)，有物件的情況下強制使角色與物件互動；為了單純處理，只讓每個房間最多含一個物件，物件包含三種：怪物(Monster)、寶箱(Chest)、商店(Shop)，在對不同物件與不同互動場景下，角色與物件的數值狀態均進行更新；互動後最重要的是檢查角色是否死亡(Is Player dead?)，是的話無需進行後續動作，直接轉移至遊戲狀態檢查。

在沒有物件或角色為死亡的情況下，會先顯示地圖(Map)，讓使用者在後續選擇移動方向較為便利，再來讓使用者選擇移動(Move)、顯示角色狀態(Player’s status)、使用背包(Use backpack)、遊戲選單(Game options)，選擇選項後會完成對應動作，再轉移至遊戲狀態檢查；遊戲選單中含有存檔(Save game)、讀檔(Load game)、離開(Quit)三個選項；存檔無論成功與否，均會轉移至遊戲狀態檢查；讀檔成功會轉至遊戲環境參數設定，失敗則轉至遊戲狀態檢查；離開是直接終止程式。

**(c) 遊戲狀態檢查：**

此功能用來判斷與控制遊戲是否結束；遊戲結束分成通關狀態與非通關狀態，前者條件為角色擊敗最終魔王(Boss)且抵達終點房間，後者為角色死亡；功能判斷遊戲結束則會終止程式，遊戲未結束則會轉移至遊戲動作功能。

**2. 程式實作細節**

- Basic Functions

- Optional Enhancement

Results

Discussion

Conclusion