TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

----- □ & □ -----



**BÁO CÁO MÔN HỌC PROJECT I**

**Đề tài: Thiết kế và tạo game Dò Mìn (Minesweeper)**

Giảng viên hướng dẫn: **Đỗ Tiến Dũng**

Nhóm sinh viên thực hiện:

|  |  |  |
| --- | --- | --- |
| STT | Họ và tên | MSSV |
| 1 | Đỗ Đặng Phương | 20194141 |
| 2 | Chu Đình Đức | 20194021 |
| 3 | Lê Mạnh Cường | 20194000 |
|  |  |  |

**Hà Nội, năm 2021**

# MỤC LỤC VÀ CÁC DANH MỤC

MỤC LỤC

[MỤC LỤC VÀ CÁC DANH MỤC 1](#_Toc91267523)

[DANH MỤC HÌNH ẢNH 3](#_Toc91267524)

[LỜI CAM KẾT 4](#_Toc91267525)

[TÓM TẮT 4](#_Toc91267526)

[1. GIỚI THIỆU 5](#_Toc91267527)

[1.1 Mục tiêu, phạm vi của đồ án 5](#_Toc91267528)

[1.2 Giải pháp 5](#_Toc91267529)

[1.3 Bố cục của đồ án: 5](#_Toc91267530)

[1.4 Phân công công việc trong nhóm đồ án 6](#_Toc91267531)

[2. PHÂN TÍCH YÊU CẦU 6](#_Toc91267532)

[2.1 Tổng quan chức năng 6](#_Toc91267533)

[2.2 Đặc tả chức năng (Use case đặc tả chi tiết chức năng) 7](#_Toc91267534)

[2.2.1 Điểm cao 7](#_Toc91267535)

[2.2.2 Chọn level 7](#_Toc91267536)

[2.2.3 Thông tin 8](#_Toc91267537)

[2.2.4 Thoát trò chơi 8](#_Toc91267538)

[2.2.5 Mở ô 8](#_Toc91267539)

[2.2.6 Cắm cờ 8](#_Toc91267540)

[2.2.7 Gợi ý 8](#_Toc91267541)

[2.3 Các yêu cầu khác 9](#_Toc91267542)

[3. THIẾT KẾ 9](#_Toc91267543)

[3.1 Thiết kế cơ sở 9](#_Toc91267544)

[3.2 Thiết kế chi tiết 9](#_Toc91267545)

[3.3 Cấu trúc dữ liệu và thuật toán 16](#_Toc91267546)

[4. XÂY DỰNG ỨNG DỤNG (CODING) 19](#_Toc91267547)

[4.1 Thư viện và công cụ sử dụng 19](#_Toc91267548)

[4.2 Quy tắc viết code (simple coding rule). 20](#_Toc91267549)

[5. KIỂM THỬ 21](#_Toc91267550)

[5.1 Unit test 21](#_Toc91267551)

[5.2 System test 24](#_Toc91267552)

[6. KẾT LUẬN 26](#_Toc91267553)

[6.1 Các đóng góp chính 26](#_Toc91267554)

[6.2 Hướng phát triển 27](#_Toc91267555)

[7. TÀI LIỆU THAM KHẢO 27](#_Toc91267556)

# DANH MỤC HÌNH ẢNH

[Hình 1: Bố cục đồ án 5](#_Toc91254309)

[Hình 2: Sơ đồ use-case cho chương trình minesweeper 7](#_Toc91254310)

[Hình 3: Biểu đồ mô tả quan hệ các lớp 9](#_Toc91254311)

[Hình 4: Giao diện menu 9](#_Toc91254312)

[Hình 5: Giao diện chọn level 10](#_Toc91254313)

[Hình 6: Giao diện chiến thắng 10](#_Toc91254314)

[Hình 7: Giao diện thua 11](#_Toc91254315)

[Hình 8: Giao diện bảng điểm 11](#_Toc91254316)

[Hình 9: Giao diện phần thông tin 12](#_Toc91254317)

[Hình 10: Giao diện chức năng thoát 12](#_Toc91254318)

[Hình 11: Lớp console 13](#_Toc91254319)

[Hình 12: Lớp functions 13](#_Toc91254320)

[Hình 13: Lớp main.h 14](#_Toc91254321)

[Hình 14: Lớp main.cpp 14](#_Toc91254322)

[Hình 15: Lớp resource.h 15](#_Toc91254323)

[Hình 16: Các file âm thanh 15](#_Toc91254324)

[Hình 17: Các file lưu điểm cao và thông tin 16](#_Toc91254325)

[Hình 18: Hàm sinh bom ngẫu nhiên 17](#_Toc91254326)

[Hình 19: Hàm đếm bom lân cận 18](#_Toc91254327)

[Hình 20: Hàm mở ô 19](#_Toc91254328)

[Hình 21: Các thư viện được sử dụng 19](#_Toc91254329)

[Hình 22: Các hàm trong functions.h 20](#_Toc91254330)

[Hình 23: Hàm win() được viết đầy đủ trong functions.cpp 20](#_Toc91254331)

[Hình 24: Comment trong functions.cpp 21](#_Toc91254332)

[Hình 25: Comment trong functions.h 21](#_Toc91254333)

[Hình 26: Minh hoạ đặt tên biến 21](#_Toc91254334)

# LỜI CAM KẾT

Nhóm Project I gồm 3 sinh viên Đỗ Đặng Phương, Chu Đình Đức và Lê Mạnh Cường cam kết trung thực trong quá trình làm bài, mã nguồn là tự viết và có tham khảo một số tài liệu trên mạng.

# TÓM TẮT

Giới thiệu vấn đề: Từ định hướng môn học là các ứng dụng trên Windows nhóm đã chọn mục tiêu của đồ án lần này là thiết kế và tạo trò chơi Dò Mìn.

Lý do chọn đề tài: Game Dò Mìn là một cực kỳ quen thuộc trên máy tính vài năm về trước. Hiện nay, nhiều người vẫn còn sở thích chơi game Dò mìn trên các trang web online của game bởi game đòi hỏi tính tư duy, logic khá cao đề giải quyết trò chơi. Đòi hỏi người chơi cần biết suy đoán vị trí của trái bom để tránh mở ra và chiến thắng trò chơi. Ngoài ra game cũng đem lại thời gian giải trí sau những giờ làm việc căng thẳng, Vì vậy, nhóm em đã quyết định lựa chọn là làm game Dò mìn.

Giải pháp thực hiện đồ án: Chương trình sẽ được sử dụng ngôn ngữ C, C++ và thực hiện trên ứng dụng Microsoft Visual Studio 2019 trên giao diện console.

Kết quả đạt được: Nhóm đã thực hiện đầy đủ các chức năng mà nhóm đề ra. Trò chơi có thể chơi được với các mức khác nhau, xem thông tin, có thể lưu và xem điểm cao khi chiến thắng trò chơi.

# 1. GIỚI THIỆU

## 1.1 Mục tiêu, phạm vi của đồ án

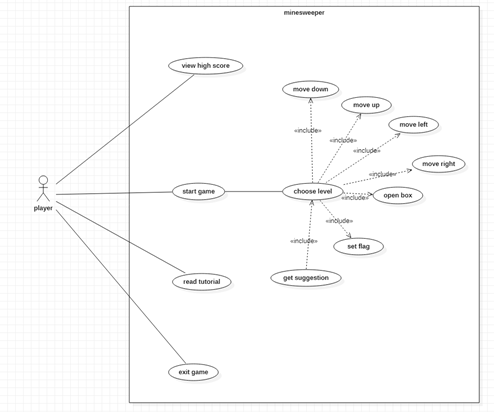
**Mục tiêu của đồ án**: thiết kế tạo trò chơi dò mìn - minesweeper trên windows.

**Phạm vi của đồ án:** chương trình được xây dựng cho các máy tính có hệ điều hành windows và được tải về hoặc sử dụng khi có sự đồng ý của tất cả thành viên nhóm.

## 1.2 Giải pháp

Sử dụng ngôn ngữ lập trình C++ và các thư viện hỗ trợ.

## 1.3 Bố cục của đồ án:



Hình 1: Bố cục đồ án

Chương trình được nhóm em chia thành các module sau:

**Module console.h**

Gồm các hàm thao tác với console như:

* hàm tô màu nền
* tô màu chữ
* thay đổi kích cỡ màn hình console
* xóa con trỏ chuột
* di chuyển con trỏ đến tọa độ (x, y) màn hình console

**Module console.cpp**

Định nghĩa các hàm trong console.h.

**Module function.h**

Các hàm xử lý game:

* Tạo mảng 2 chiều và xóa mảng 2 chiều
* Các hàm khởi tạo mức chơi, khởi tạo bom ngẫu nhiên
* Xử lý sự kiện nhập từ bàn phím
* Xử lý thắng thua
* Hàm vẽ bảng, vẽ ô
* Vẽ menu chính
* Hàm xử lý sự kiện chuột
* Hàm xuất ra thông tin trò chơi
* Hiển thị trạng thái chơi game ( thắng thua hay đang chơi)
* Lưu thời gian hoàn thành tương ứng với mỗi mức ( cho phần view high score : cho đọc khi xem và cho viết khi hoàn thành giải ma trận)

**Module function.cpp**

Định nghĩa các hàm của function.h.

**Module resource.h**

Chứa các tệp âm thanh hoặc hình ảnh có thể dùng trong game.

**Module main.h**

Định nghĩa các cấu trúc bảng, cấu trúc ô và một số phần khác.

**Module main.cpp**

Chạy chương trình chính bằng cách gọi các hàm ở các module khác.

## 1.4 Phân công công việc trong nhóm đồ án

. Đỗ Đặng Phương - 20194141 : thiết kế giao diện game, tạo thông tin game, xử lý chuột, làm báo cáo.

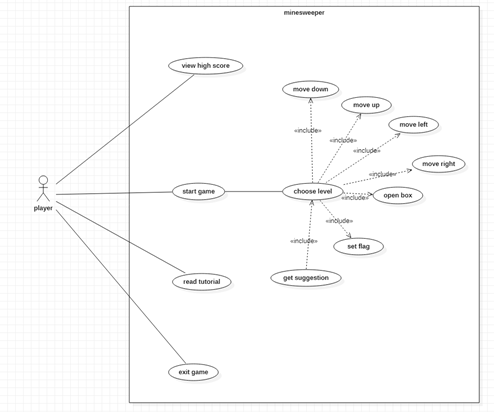
. Chu Đình Đức - 20194021 : viết các hàm xử lý game : mở ô, tạo mảng hai chiều, tạo random bom, cắm cờ, làm báo cáo.

. Lê Mạnh Cường - 20194000: viết các hàm xử lý thao tác phím, cùng xử lý chuột với bạn Phương, lưu điểm, làm cáo cáo.

# 2. PHÂN TÍCH YÊU CẦU

## 2.1 Tổng quan chức năng

* Mô tả các chức năng: các chức năng chính bao gồm chọn level, chơi game, điểm cao, thông tin.



Hình 2: Sơ đồ use-case cho chương trình minesweeper

* Quy trình nghiệp vụ liên quan: sau khi chọn cấp độ và chơi game, nếu win thì sẽ lưu điểm hoặc thoát, nếu thua thì sẽ chơi lại hoặc thoát.

## 2.2 Đặc tả chức năng (Use case đặc tả chi tiết chức năng)

### 2.2.1 Điểm cao

* Tên use case: view high-score
* Mục đích: xem các thành tích cao nhất đã đạt được trước đó - thời gian ngắn nhất giải xong 1 ma trận.
* Tóm lược: Chứa bảng các thành tích tốt nhất đã đạt được ở từng loại ma trận
* Mô tả kịch bản: người chơi sử dụng phím để chọn lựa chọn trong menu, hệ thống sẽ hiển thị bảng thành tích lên và khi xem xong người chơi có thể quay lại trang chủ bằng nút return trong bảng.

### 2.2.2 Chọn level

* Tên use case: level
* Mục đích: chọn các mức chơi ứng với ma trận tương ứng.
* Tóm lược: Chứa 3 lựa chọn tương ứng với ba mức chơi đó là

. dễ: ma trận 9x9 với 10 quả mìn trong ma trận

. trung bình: ma trận 16x16 với 40 quả mìn trong ma trận

. khó: ma trận 24x24 với 99 quả mìn trong ma trận

* Giao diện ma trận chứa:

. ma trận mìn dưới dạng ma trận 2 chiều

. ô đếm thời gian

. nút quay lại trang chủ

* Mô tả kịch bản: người chơi sử dụng phím để chọn lựa chọn trong menu, hệ thống sẽ hiển thị bảng chọn các mức chơi và sau khi chọn thì hệ thống sẽ hiển thị các ma trận tương ứng với mức chơi đã chọn cùng với giao diện chơi. Người chơi sẽ sử dụng các phím để đánh dấu ô nào có thể là mìn, mở ô mà người chơi cho là không chứa mìn và hệ thống sẽ đếm thời gian cho đến khi người chơi giải xong ma trận hoặc mở vào ô có mìn. Sau đó người chơi sẽ được đưa quay trở lại trang chủ.

### 2.2.3 Thông tin

* Tên use case: read tutorial.
* Mục đích: chọn trong menu trang chủ để đọc hướng dẫn.
* Tóm lược: chứa phần hướng dẫn chơi và luật chơi.
* Mô tả kịch bản: người chơi sử dụng các phím để chọn trong menu, hệ thống sẽ hiển thị mục hướng dẫn chơi.

### 2.2.4 Thoát trò chơi

* Tên use case: exit game
* Mục đích:thoát trò chơi
* Mô tả kịch bản: người chơi sử dụng các phím để chọn trong menu để thoát chương trình trò chơi.

### 2.2.5 Mở ô

* Tên use case: open box
* Mục đích: mở 1 ô do người chơi chỉ định
* Tóm lược: 1 ô được mở ra cùng với các ô lân cận (nếu được phép mở)
* Mô tả kịch bản: người chơi di chuyển đến ô cần mở, ấn Z hoặc click chuột trái để mở ô đó, đồng thời các ô lân cận cũng có thể được mở nếu xung quanh ô đó không có bom (đệ quy).

### 2.2.6 Cắm cờ

* Tên use case: set flag
* Mục đích: cắm cờ để đánh dấu ô có bom
* Tóm lược: ô được cắm cờ sẽ hiển thị cắm cờ P.
* Mô tả kịch bản: người chơi di chuyển đến ô nghi ngờ có bom và cắm cờ bằng phím X hoặc click chuột phải, ô đó hiện chữ P, khi này người chơi có thể bỏ cắm cờ bằng cách ấn X hoặc chuột phải.

### 2.2.7 Gợi ý

* Tên use case: get suggestion
* Mục đích: chỉ cho người chơi những ô có bom.
* Tóm lược: 1 ô có bom được cắm cờ.
* Mô tả kịch bản: người chơi ấn S, 1 ô bom bất kỳ sẽ được cắm cờ.

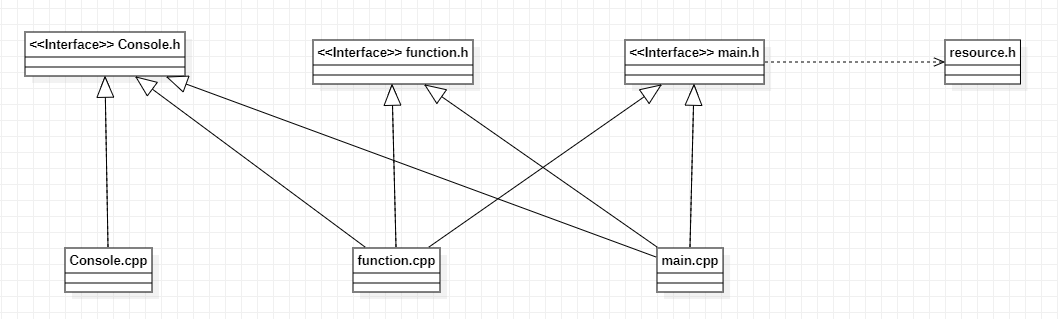
## 2.3 Các yêu cầu khác

* Âm thanh khi mở ô, cắm cờ, win, lose.

# 3. THIẾT KẾ

## 3.1 Thiết kế cơ sở

Các thành phần, gói (component, package) và tương tác giữa chúng:Biểu đồ UML mô tả quan hệ các lớp, có 8 lớp (4 file .h và 3 file.cpp).



Hình 3: Biểu đồ mô tả quan hệ các lớp

## 3.2 Thiết kế chi tiết

* Thiết kế giao diện: Giao diện được thiết kế trên màn hình console - đồ họa trên cửa sổ dòng lệnh, dựa vào thư viện Console.h.
* Cụ thể: P: cắm cờ

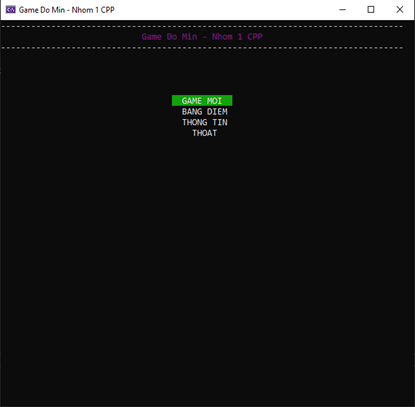
B: bom

Px: cắm cờ sai

Các số từ 1-8: số bom lân cận

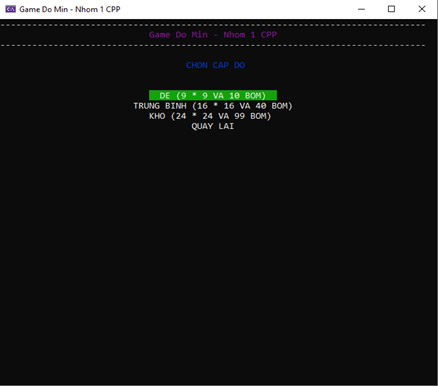
Các ô trong ma trận: màu xám - trắng xen kẽ

* MENU GAME



Hình 4: Giao diện menu

* GAME MOI → CHON CAP DO



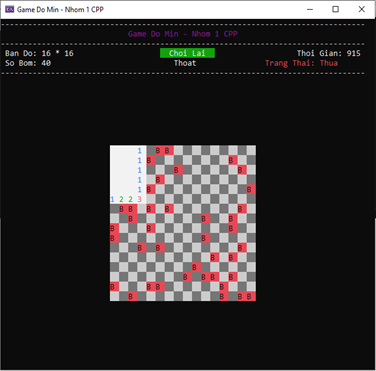
Hình 5: Giao diện chọn level

* WIN



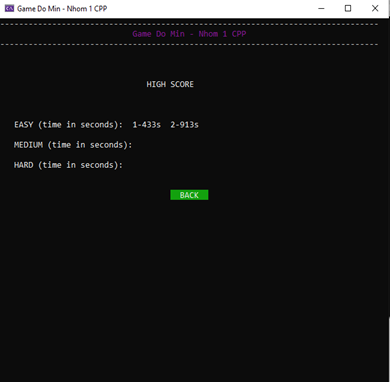
Hình 6: Giao diện chiến thắng

* LOSE



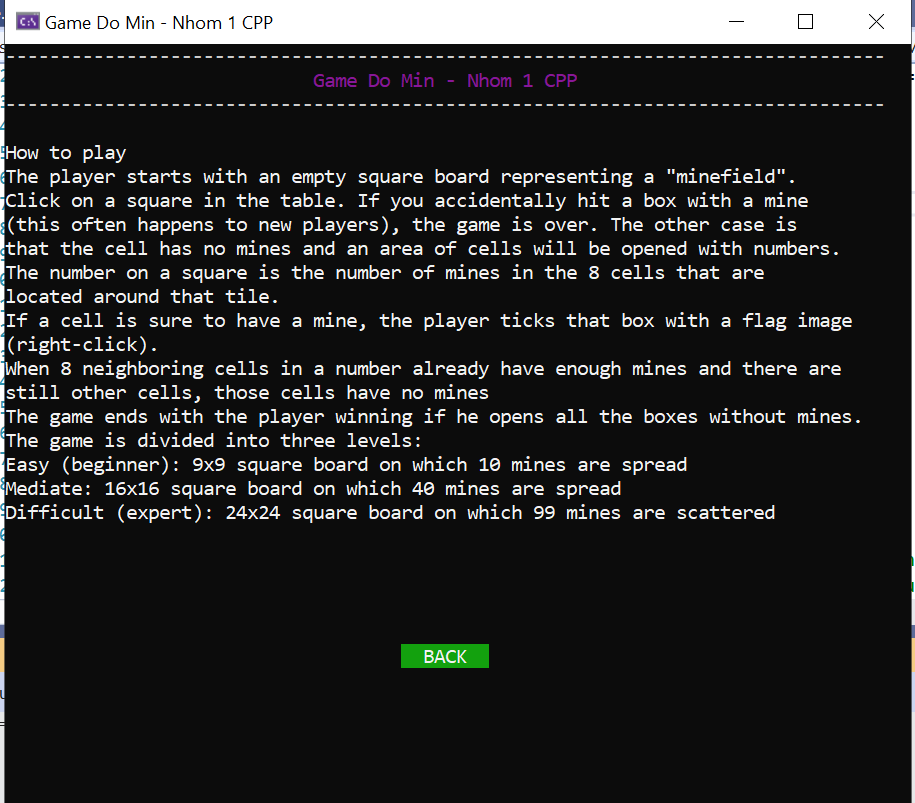
Hình 7: Giao diện thua

* BANG DIEM



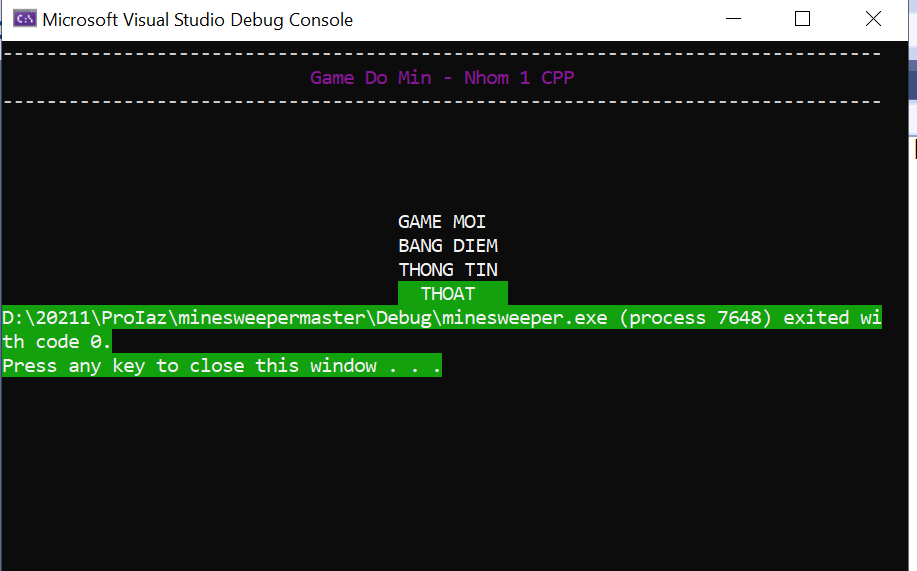
Hình 8: Giao diện bảng điểm

* THONG TIN



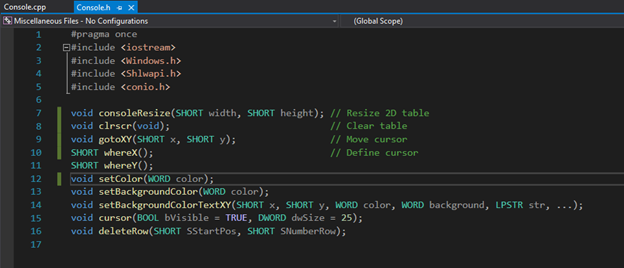
Hình 9: Giao diện phần thông tin

* THOAT



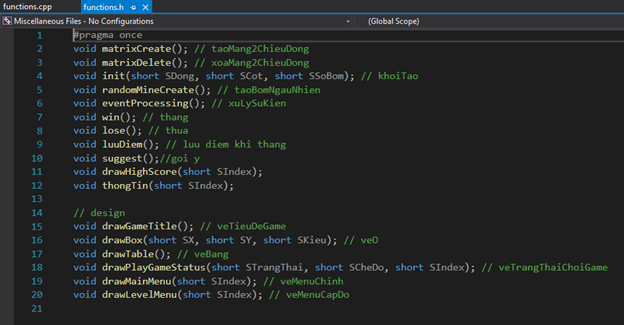
Hình 10: Giao diện chức năng thoát

* Thiết kế lớp
* Lớp Console.h và Console.cpp:



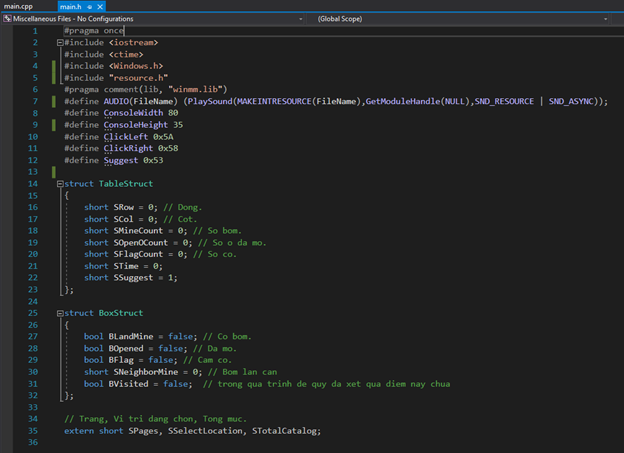
Hình 11: Lớp console

* Lớp functions.h và functions.cpp

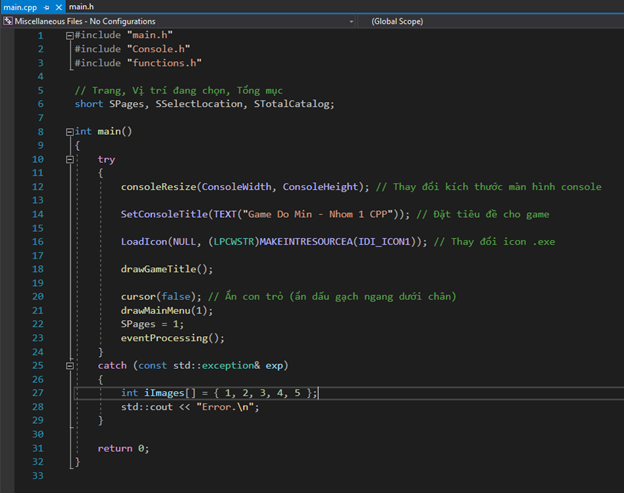


Hình 12: Lớp functions

* Lớp main.h và main.cpp

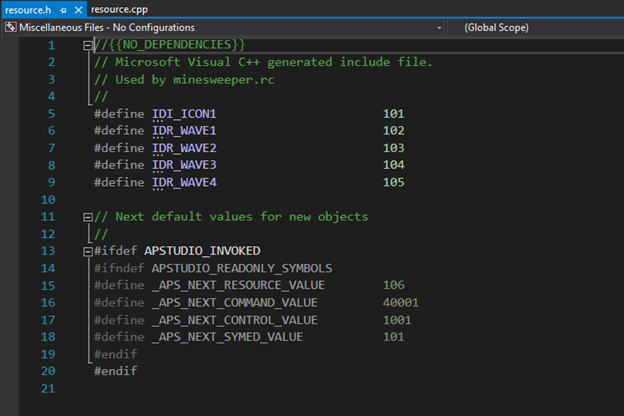


Hình 13: Lớp main.h



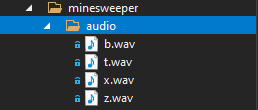
Hình 14: Lớp main.cpp

* Lớp resource.h



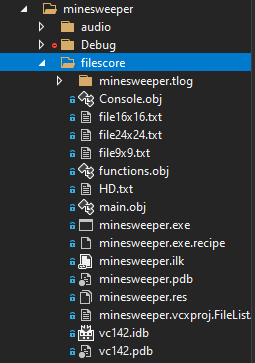
Hình 15: Lớp resource.h

* Thiết kế cơ sở dữ liệu
* Âm thanh: gồm 4 âm thanh (mở ô, đặt cờ, bom nổ, win) bằng cách chạy file .wav bằng thư viện Window.h.



Hình 16: Các file âm thanh

* File thông tin: file HD.txt
* File lưu điểm cao: file9x9.txt, file16x16.txt, file24x24.txt



Hình 17: Các file lưu điểm cao và thông tin

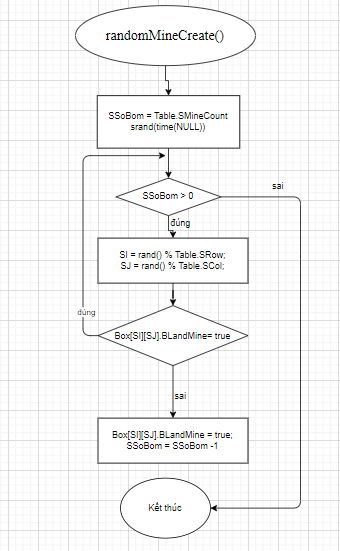
## 3.3 Cấu trúc dữ liệu và thuật toán

Thuật toán sinh ma trận ngẫu nhiên:

* randomMineCreate():

. cập nhật số bom sau mỗi lần random: SSoBom

. khi số bom > 0 ta vẫn đặt bom trong mảng hai chiều bằng cách random dòng(Table.SRow), cột(Table.SCol) - mỗi lần chơi sẽ là mỗi lần random vị trí khác nhau với hàm srand(time(NULL)), nếu vị trí đó trong Table đã có bom thì ta bỏ qua và random vị trí khác, nếu chưa có bom thì cắm cờ Bom (BLandMine) tại vị trí đó, số lượng Bom cần cắm sẽ giảm sau mỗi lần random được.



Hình 18: Hàm sinh bom ngẫu nhiên

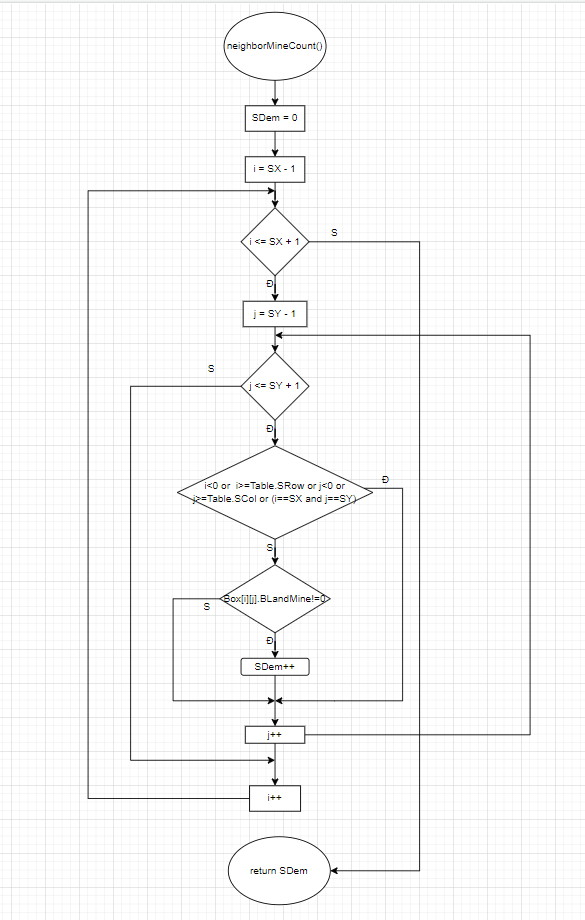
Thuật toán loang ma trận (Đếm số bom lân cận, giải thuật đệ quy) :

* Đếm số Bom lân cận vị trí xác định (neighborMineCount(short SX, short SY)):

. SX,SY là vị trí cần xét được truyền vào bằng tham số.

. Biến đếm SDem để đếm số bom lân cận, ta sẽ duyệt các vị trí trong mảng lân cận với vị trí đang xét theo hình vuông với vị trí đang xét là tâm, cạnh là 3 ô, nếu ô lân cận vẫn trong mảng và không phải là vị trí SX,SY và đang có bom(cờ BLandMine = true) thì biến đếm sẽ tăng thêm 1.

. hàm trả về biến đếm SDem.

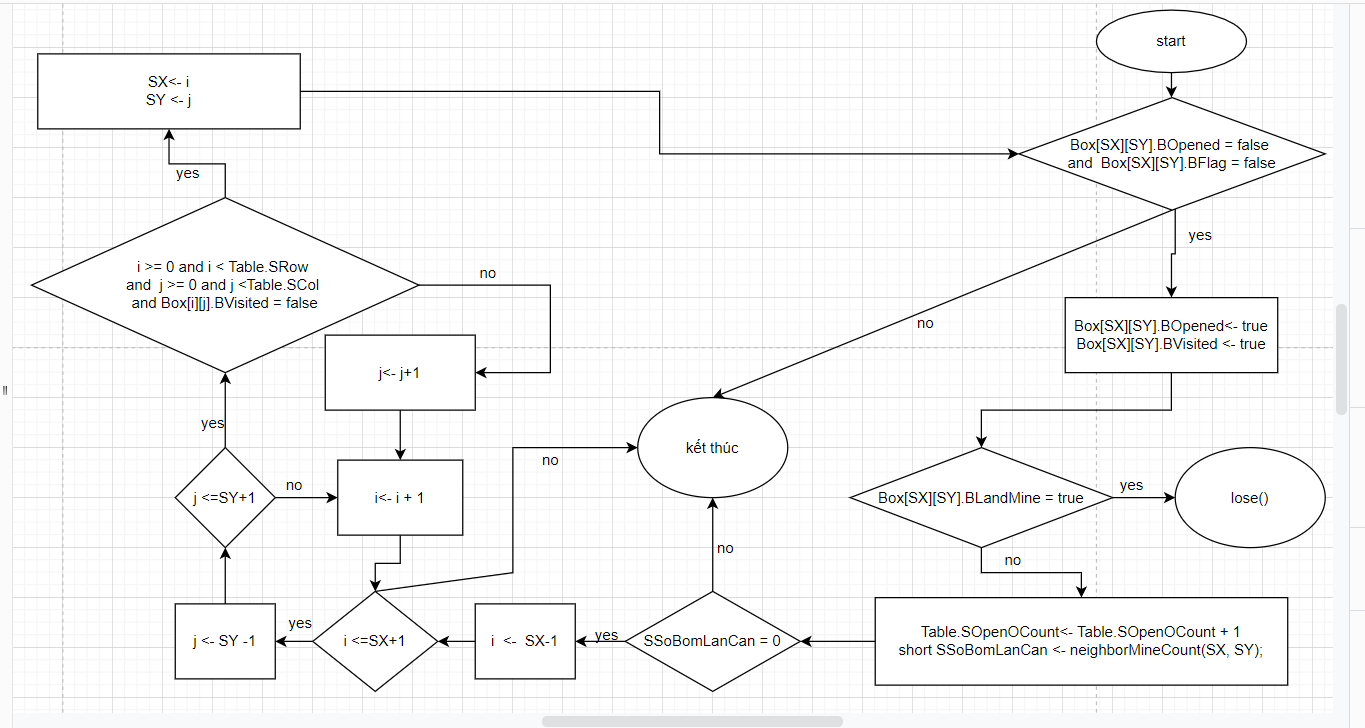


Hình 19: Hàm đếm bom lân cận

* Mở ô ( boxOpen(SX,SY)):

. Xét vị trí được chọn mở:

. Nếu vị trí đó có cờ Bom (BLandMine = true) thì tính thua, gọi tới trang thua lose(). Nếu vị trí đó chưa cắm cờ, chưa được mở thì mở ô thì tiếng hành mở ô (BOpened = true) và cờ đã xét (BVisited = true) , đồng thời tiến hành đếm số bom lân cận, nếu có bom lân cận thì hiển thị số bom, nếu số bom lân cận bằng 0 thì tiếp tục gọi tới hàm mở ô đối với các ô xung quanh có bVisited = false.



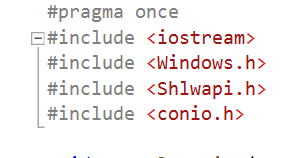
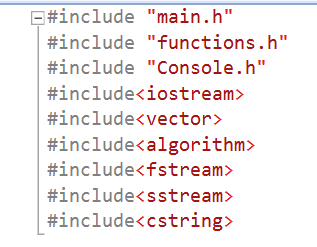
Hình 20: Hàm mở ô

# 4. XÂY DỰNG ỨNG DỤNG (CODING)

## 4.1 Thư viện và công cụ sử dụng

Thư viện:

Chương trình sử dụng các thư viện của C, C++ bao gồm <windows.h>, <conio.h>, <Shlwapi.h>, <ctime>, <sstream>, <cstring>, <fstream>, <algorithm>, <vector>.



Hình 21: Các thư viện được sử dụng

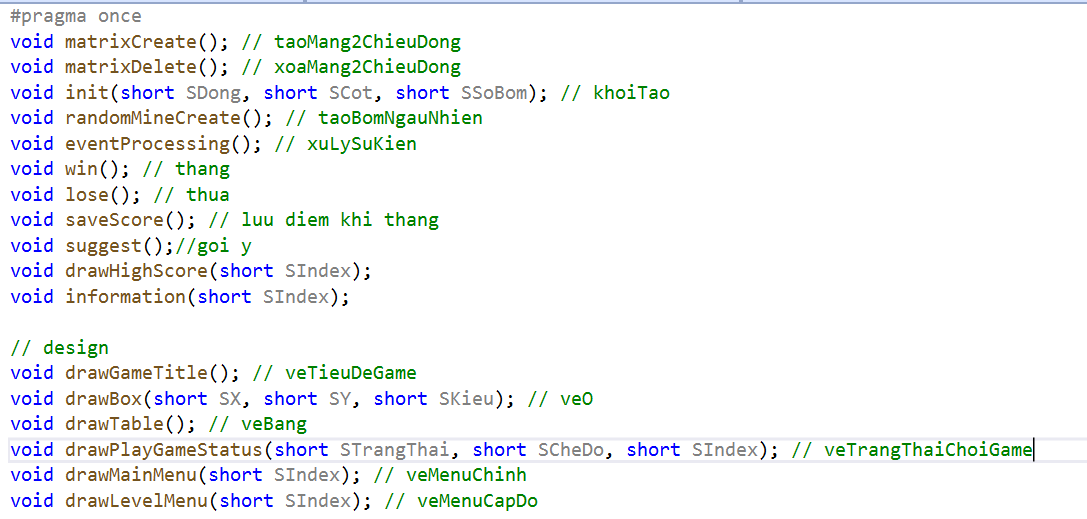
Ngoài ra chương trình còn sử dụng một số hàm mở rộng thao tác với console được cộng đồng coder xây dựng (trong console.h).

Công cụ sử dụng:

Chương trình được chạy trên Microsoft Visual Studio 2019. Ngoài ra có thể chạy trên các phiên bản khác của Microsoft Visual Studio.

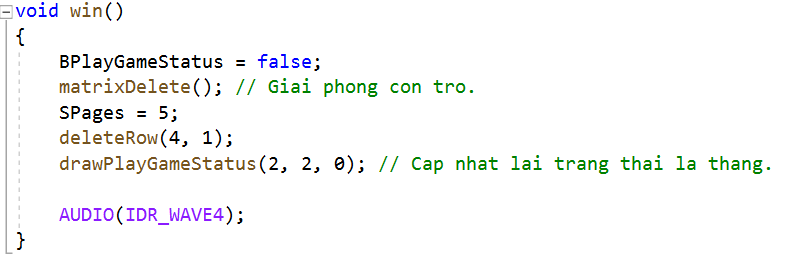
## 4.2 Quy tắc viết code (simple coding rule).

* Tên các hàm, các cấu trúc tổng quát (cấu trúc bảng, cấu trúc ô) sẽ được viết trong các file header (các file .h).



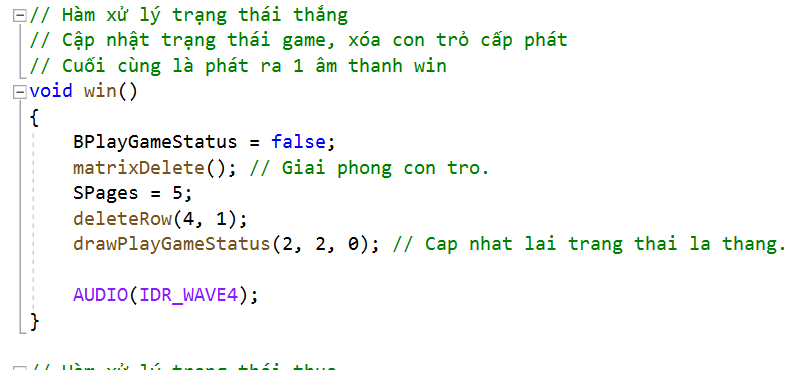
Hình 22: Các hàm trong functions.h

* Các hàm sẽ được viết đầy đủ trong các file functions.cpp và console.cpp. Chương trình chính được viết trong file main.cpp bằng cách sử dụng các hàm đã được định nghĩa trước đó.

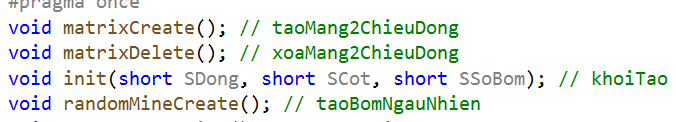


Hình 23: Hàm win() được viết đầy đủ trong functions.cpp

* Quy tắc comment và viết hàm, biến:
  + Các comment giải thích hàm ở console.cpp và functions.cpp sẽ được viết trước khi định nghĩa hàm. Ở các file header (file .h) viết comment ngắn gọn cùng dòng với hàm.

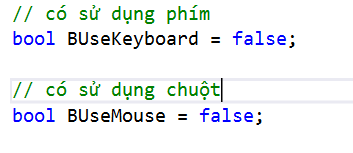


Hình 24: Comment trong functions.cpp



Hình 25: Comment trong functions.h

* + Trong lúc viết hàm có thể thêm comment để giải thích câu lệnh, thuật toán cho dễ hiểu hơn.
  + Với các biến toàn cục cũng được comment tương tự như vậy. Ngoài ra kiểu biến có thể ảnh hưởng đến ký tự đầu tiên của biến.
    - Ví dụ một biến kiểu short có thể đặt bắt đầu bằng chữ s (SX) hay một biến kiểu int có thể bắt đầu bằng chữ I (Itime), kiểu bool thì bắt đầu bằng chữ B.



Hình 26: Minh hoạ đặt tên biến

# 5. KIỂM THỬ

## 5.1 Unit test

* Test case cho các module, function, class

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Tên hàm** | **Mô tả test case** | **Kết quả dự đoán** | **Kết quả thực** | **Pass /Fail** |
| 1 | draw Table() | vẽ mảng hai chiều tương ứng với các mức chơi 9x9, 16x16, 24x24, các ô màu đan xen . | tạo được mảng hai chiều với các ô màu đan xen trắng, xám dưới tiêu đề và trạng thái game | mảng hai chiều với các ô màu đan xen trắng, xám dưới tiêu đề và trạng thái game | pass |
| 2 | drawBox() | vẽ các ô trong mảng hai chiều với kích thước vuông, màu đan xen . | các ô trong mảng hai chiều với kích thước vuông, màu đan xen | các ô trong mảng hai chiều với kích thước vuông, màu đan xen | pass |
| 3 | click Right() | cắm cờ vào ô mà người chơi cho là có bom | hiện chữ P tại ô được đánh dấu | hiện chữ P tại ô được đánh dấu | pass |
| 4 | clickLeft() | mở ô mà người chơi cho là không có bom | hiện số nếu ô đó không có bom, hiện giao diện thua nếu ô đó có bom | hiện số vì ô đó không có bom, hiện giao diện khi ô được mở có bom | pass |
| 5 | openBox() | mở ô khi bấm chuột trái hoặc phím z khi đang chơi | hiện số hoặc chữ P - có bom tại ô được mở, nếu xung quanh không có bom thì mở tiếp các ô xung quanh đó với thuật toán đệ quy | hiện số - không có bom, hiện P khi không có bom, các ô lân cận không có bom thì cũng được mở | pass |
| 6 | draw Play  Game Status  () | vẽ trạng thái khi chơi game | hiện trạng thái chơi, số bom, số gợi ý còn lại, thời gian chơi, mức chơi, số bom còn lại dưới thanh tiêu đề game | hiện trạng thái chơi, số bom, số gợi ý còn lại, thời gian chơi, mức chơi, số bom còn lại dưới thanh tiêu đề game | pass |
| 7 | mouse Processing() | nhận và xử lý thao tác chuột của người chơi khi chơi | khi đang chơi: bấm chuột trái thì sẽ mở ô, chuột phải sẽ cắm cờ khi người chơi click vào ô trong mảng hai chiều đã được tạo | bấm chuột trái thì sẽ mở ô, chuột phải sẽ cắm cờ khi người chơi click vào ô trong mảng hai chiều đã được tạo | pass |
| 8 | keyBoard  Processing() | nhận và xử lý thao tác từ bàn phím của người chơi | nút di chuyển lên thì con trỏ sẽ đi lên, khi lên đến tọa độ trên cùng sẽ chuyển xuống dưới cùng, tương tự các phím di chuyển xuống, sang trái, sang phải.  Khi chơi: z - mở ô, x - cắm cờ,  Esc - quay về trang menu  Enter: chọn | nút di chuyển lên thì con trỏ sẽ đi lên, khi lên đến tọa độ trên cùng sẽ chuyển xuống dưới cùng, tương tự các phím di chuyển xuống, sang trái, sang phải.  Khi chơi: z - mở ô, x - cắm cờ,  Esc - quay về trang menu  Enter: chọn | pass |
| 9 | information  () | đọc thông tin từ file HD.txt | hiện thông tin về game theo từng dòng | hiện thông tin về game theo từng dòng | pass |
| 10 | saveScore() | lưu thời gian chơi khi người chơi chơi hết một vấn game | lưu thời gian chơi vào file9x9.txt, file16x16.txt, file24x24.txt tương ứng các mức chơi nếu thời gian đó nằm trong top 6 ngắn nhất | lưu thời gian chơi vào file9x9.txt, file16x16.txt, file24x24.txt tương ứng các mức chơi nếu thời gian đó nằm trong top 6 ngắn nhất | pass |
| 11 | draw  HighScore() | xem thông tin bảng điểm các mức chơi | hiện thời gian nằm trong top 6 ngắn nhất tương ứng trong mỗi file file9x9.txt, file16x16.txt, file24x24.txt | hiện thời gian nằm trong top 6 ngắn nhất tương ứng trong mỗi file file9x9.txt, file16x16.txt, file24x24.txt | pass |

## 5.2 System test

* Test case cho chương trình, các chức năng

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Tên chức năng** | **Mô tả chức năng** | **Kết quả dự đoán** | **Kết quả thực** | **Pass/Fail** |
| 1 | GAME MOI | chức năng này để chuyển sang trang chọn mức game để chơi | chuyển sang trang chọn mức chơi: hiển thị 3 mức chơi dễ, trung bình, khó, và lựa chọn quay lại trang menu | chuyển sang trang chọn mức chơi: hiển thị 3 mức chơi dễ, trung bình, khó, và lựa chọn quay lại trang menu | pass |
| 2 | BANG DIEM | chức năng này xem bảng điểm : top 6 điểm cao trong mỗi mức chơi | hiển thị top 6 điểm cao trong mỗi mức chơi dễ - 9x9, trung bình -16x16, khó- 24x24 và lựa chọn quay lại | hiển thị top 6 điểm cao trong mỗi mức chơi dễ - 9x9, trung bình -16x16, khó- 24x24 và lựa chọn quay lại | pass |
| 3 | THONG TIN | chức năng này để xem thông tin về game minesweeper | hiển thị thông tin về game minesweeper và lựa chọn quay lại | hiển thị thông tin về game minesweeper và lựa chọn quay lại | pass |
| 4 | THOAT | chức năng này để thoát game, đóng cửa sổ game | đóng cửa sổ game | đóng cửa sổ game | pass |
| 5 | DE(mức chơi) | chức năng này để vào chơi mức dễ với bảng 9x9, 10 bom, 2 gợi ý | hiển thị cửa sổ chơi với tiêu đề game, trạng thái game, bảng 9x9 để người chơi thao tác | hiển thị cửa sổ chơi với tiêu đề game, trạng thái game, bảng 9x9 để người chơi thao tác | pass |
| 6 | TRUNG BINH(mức chơi) | chức năng này để vào chơi mức dễ với bảng 16x16,40 bom, 5 gợi ý | hiển thị cửa sổ chơi với tiêu đề game, trạng thái game, bảng 16x16 để người chơi thao tác | hiển thị cửa sổ chơi với tiêu đề game, trạng thái game, bảng 16x16 để người chơi thao tác | pass |
| 7 | KHO(mức chơi) | chức năng này để vào chơi mức dễ với bảng 9x9, 99 bom, 9 gợi ý | hiển thị cửa sổ chơi với tiêu đề game, trạng thái game, bảng 24x24 để người chơi thao tác | hiển thị cửa sổ chơi với tiêu đề game, trạng thái game, bảng 24x24 để người chơi thao tác | pass |
| 8 | LUU DIEM | chức năng này để lưu điểm khi người chơi đã giải hết số ô không chứa bom | lưu thời gian chơi vào file9x9.txt, file16x16.txt, file24x24.txt tương ứng các mức chơi nếu thời gian đó nằm trong top 6 ngắn nhất | lưu thời gian chơi vào file9x9.txt, file16x16.txt, file24x24.txt tương ứng các mức chơi nếu thời gian đó nằm trong top 6 ngắn nhất | pass |
| 9 | CHOI LAI | chức năng này để chơi lại mức game vừa chơi thua | tạo lại mảng hai chiều để người chơi có thể chơi lại từ đầu, các trạng thái game được reset về như lúc bắt đầu | tạo lại mảng hai chiều để người chơi có thể chơi lại từ đầu,các trạng thái game được reset về như lúc bắt đầu | pass |
| 10 | BACK | chức năng này để quay lại trang menu khi đang đọc thông tin hoặc xem điểm cao | quay lại trang menu | quay lại trang menu | pass |
| 11 | QUAY LAI | chức năng này để quay lại trang menu khi đang ở trang chọn mức chơi | quay lại trang menu | quay lại trang menu | pass |
| 12 | Phím Esc | chức năng này để quay lại trang menu khi người chơi muốn | quay lại trang menu | quay lại trang menu | pass |

# 6. KẾT LUẬN

## 6.1 Các đóng góp chính

* Kết quả thu được: Nhóm đã thực hiện đề tài tương đối hoàn chỉnh so với mục tiêu đề ra từ trước. Về chức năng chính, chương trình có thể chơi được với các mức khác nhau, sử dụng được cả chuột lẫn bàn phím để chơi. Ngoài ra còn một số chức năng khác như xem thông tin hay bảng điểm đã nêu ở các phần trước.
* Nhận xét về tính khả thi, tính chính xác của kết quả, tính thực tế của đồ án: Đề tài lựa chọn phù hợp với một trong các định hướng của môn học đó là ứng dụng trên Windows. Kết quả thu được có hoạt động ổn định. Người chơi dễ dàng sử dụng chuột hoặc bàn phím để chơi. Trò chơi mang lại tính giải trí cho người chơi.
* Đối với mỗi sinh viên trong nhóm: Qua quá trình thực hiện đồ án, mỗi thành viên cũng biết thêm kiến thức về C, C++ và hiểu hơn về cách xây dựng một ứng dụng cụ thể. Mỗi thành viên trong nhóm đã học được nhiều kỹ năng lập trình (quy ước lập trình, gỡ rối và kiểm thử) cũng như các kỹ năng mềm (kỹ năng làm việc nhóm, kỹ năng làm báo cáo, tự tìm kiếm tài liệu, …).
* Các vấn đề gặp phải: Trong quá trình hoàn thành đồ án nhóm gặp phải một số vấn đề liên quan đến kiến thức chuyên môn (cách xử lý chuột, xử lý phím), tuy nhiên cuối cùng nhóm đã giải quyết được.

## 6.2 Hướng phát triển

Chương trình được viết trên giao diện Console vì vậy đồ hoạ có thể chưa được đẹp mắt. Vì vậy có thể phát triển bằng cách sử dụng các thư viện đồ họa nâng cao của C, C++ như SDL để làm cho giao diện trở nên đẹp hơn cũng như việc responsive của trò chơi.

Ngoài ra còn có thể một số chức năng khác như lưu tên khi chiến thắng, có thể tiếp tục trò chơi khi bấm ESC, …

# 7. TÀI LIỆU THAM KHẢO

Các kiểu dữ liệu trong C, C++: [Windows Data Types](https://docs.microsoft.com/en-us/windows/win32/winprog/windows-data-types)

Console structures: [Console Structures](https://docs.microsoft.com/en-us/windows/console/console-structures)

Xử lý sự kiện từ bàn phím: [Key Event Record](https://docs.microsoft.com/vi-vn/windows/console/key-event-record-str)

Xử lý sự kiện từ chuột: [Mouse Event Record](https://docs.microsoft.com/vi-vn/windows/console/mouse-event-record-str)

Các hàm thao tác với console: [Hàm thao tác trên Console](https://blogkhanhtoan.wordpress.com/2016/03/07/mot-so-ham-mo-rong-trong-cc/?fbclid=IwAR3RkDV9FYOrgbaCfqD-7xCmU001ca49hZrZGQNmCbg7b-qJrHnMOG97UoU)

Hướng dẫn C++: <https://www.w3schools.com/cpp/>