

Trường Đại học Khoa Học Tự Nhiên - ĐHQG HCM

Khoa Công nghệ thông tin - Lớp 22CLC0A



BÁO CÁO ĐỒ ÁN
HUNTING SNAKE GAME
MÔN: KỸ THUẬT LẬP TRÌNH

Sinh Viên	Võ Ngọc Gia Bảo	ID: 21127016
	Trần Nguyên Huân	ID: 21127050
	Nguyễn Minh Thông	ID: 21127696
	Nguyễn Phú Trọng	ID: 21127708
Giảng viên hướng dẫn	Thầy Trương Toàn Thịnh Thầy Hồ Tuấn Thanh Thầy Mai Anh Tuấn Thầy Nguyễn Lê Hoàng Dũng	

TP. HCM, 08, 2023

1. Thông tin thành viên nhóm

Họ tên	MSSV
Võ Ngọc Gia Bảo	21127016
Trần Nguyên Huân	21127050
Nguyễn Minh Thông	21127696
Nguyễn Phú Trọng	21127708

2. Phân công việc nhóm và đánh giá mức độ hoàn thành

Tên công việc	SV	Tỷ lệ
1. Tạo lập nhân vật - rắn và khả năng di chuyển của rắn.	Gia Bảo	100%
2. Xử lý thua khi rắn chạm tường, chạm chướng ngại vật.	Gia Bảo	100%
3. Tạo lập vật phẩm tăng điểm, hiệu ứng có lợi và hiệu ứng xấu cho người chơi.	Gia Bảo	95%
4. Khởi tạo chức năng tăng điểm cho người chơi khi chạm vật phẩm tăng điểm.	Minh Thông	100%
5. Tạo lập vùng bảng điểm, hiệu ứng tăng điểm người chơi.	Minh Thông	100%
6. Thiết kế, xây dựng khung sườn trò chơi. Điều chỉnh bố cục giao diện game.	Nguyên Huân	100%
7. Thiết kế các giao diện màn chơi trong game - Mode 2,3.	Phú Trọng	95%
8. Xây dựng thanh chức năng khi người chơi tạm dừng.	Phú Trọng	100%
9. Xây dựng chức năng New Game, Continue trong game.	Nguyên Huân	95%
10. Xây dựng chức năng Option, Save Game và Load Game.	Nguyên Huân	100%
11. Thiết kế giao diện menu chính của game.	Minh Thông	95%
12. Thiết kế, vẽ các chi tiết phụ, hình ảnh cho game.	Minh Thông	95%
13. Xây dựng chức năng Load Game.	Nguyên Huân	100%
14. Xây dựng chức năng Setting, LeaderBoard và Thông tin nhóm.	Gia Bảo	100%

1. Kịch bản trò chơi

- Khi bắt đầu trò chơi, người chơi sẽ chọn một trong bốn chế độ bao gồm:
 - + Chế độ chơi đơn - vô tận
 - + Chế độ chơi đơn - phiêu lưu
 - + Chế độ chơi đội - phiêu lưu
 - + Chế độ đối kháng giữa 2 người chơi
- Người chơi dùng các phím di chuyển (A, S, D, W hoặc tùy chỉnh) để điều khiển rắn của mình ăn quả và tránh các chướng ngại vật để tích lũy điểm số cao nhất có thể hoặc để qua màn và chiến thắng.
- Nếu người chơi không may chạm phải vật cản thì sẽ dẫn đến thua cuộc. Sau đó có thể lưu lại thông tin của người chơi.
- Tiếp tục chơi bằng cách chọn “New Game”, muốn thoát thì sẽ chọn “Return Menu”.

2. Tính năng chính

a. Chế độ chơi

- Chế độ chơi đơn - vô tận:
 - Người chơi sẽ di chuyển rắn để ăn quả và tích lũy điểm cao nhất có thể.
- Chế độ chơi đơn - phiêu lưu:
 - Người chơi sẽ di chuyển rắn tránh né các vật cản và quả độc, ăn các quả tính điểm để tích lũy điểm số và qua màn.
- Chế độ chơi đội - phiêu lưu:
 - Gồm có hai người chơi, một người điều khiển rắn xanh và một người điều khiển rắn đỏ.
 - Cả hai người chơi phải cùng nhau tích lũy điểm số, tránh chạm vào vật cản và rắn của nhau để qua màn.
- Chế độ đối kháng giữa 2 người chơi:
 - Gồm có hai người chơi đối đầu với nhau, ai giành được nhiều điểm hơn thì sẽ chiến thắng.
 - Nếu hai rắn chạm vào thân nhau thì người có nhiều điểm hơn sẽ chiến thắng.

b. Giao diện người dùng

- Sử dụng các thư viện liên quan đến màn hình (Windows.h) để khóa cố định kích thước cửa sổ chơi game, làm cho người chơi không thể thay đổi kích thước cửa sổ, làm hỏng bố cục của trò chơi. Một số hàm được sử dụng như:
 - SetWindowSize, Fix Console Window: đặt kích thước cho màn hình.
 - resize Console: thay đổi kích thước màn hình.

- Sử dụng các hàm liên quan đến tọa độ của con trỏ console để tạo ra bố cục hợp lý cho người chơi có thể dễ dàng thao tác. Một số hàm được sử dụng như:
 - GotoXY: di chuyển con trỏ đến vị trí x, y.
 - whereX: lấy tọa độ x của con trỏ.
 - whereY: lấy tọa độ y của con trỏ.
 - no CursorType: xóa con trỏ nháy.
- Các giao diện chính trong trò chơi:
 - Menu chính: bao gồm New game, Load game, Option, About, Exit.
 - + New game: Chọn một trong 4 chế độ của trò chơi để chơi.
 - + Load game: Lựa chọn mode, sau đó lựa chọn tên người chơi để load lại game.
 - + Option: Dùng để bật, tắt âm thanh, sửa đổi bàn phím theo nhu cầu.
 - + About: Trang thông tin của nhóm.
 - + Exit: Dùng để thoát trò chơi.
 - Màn hình chơi trò chơi: nơi diễn ra tất cả các sự kiện của trò chơi như di chuyển rắn, ăn quả, vật cản, cổng qua màn,...
 - Menu Pause: xuất hiện khi người chơi bấm phím 'P' khi đang chơi. Bao gồm option, guide và score.
 - + Option: dùng để vào menu chứa resume, newgame, option bật âm thanh, đổi phím, save game hoặc exit.
 - + Guide: vào menu hướng dẫn chơi.
 - + Score: hiển thị điểm số của người chơi.
 - Màn hình thắng - thua: xuất hiện khi người chơi thắng hoặc trò chơi.

c. Hiệu ứng hình ảnh

- Để các hình ảnh của trò chơi có thể chuyển động mượt mà và tách biệt với nhau thì cần sử dụng đa luồng (thư viện thread). Một luồng dùng để điều khiển rắn và một luồng để thực hiện các sự kiện của game.
- Ngoài ra còn bổ sung các hiệu ứng như vẽ hình rắn, chạy điểm số khi thua cuộc.
- Trong màn hình menu, khi người chơi nhấn các phím lên xuống để lựa chọn thì các hình ảnh rắn của dòng đang được chọn sẽ dài ra, còn các dòng khác sẽ ngắn hơn.
- Trong các giao diện menu thì dòng mà đang được lựa chọn sẽ có màu chữ khác với các dòng còn lại.
- Đối với new game hay load game thì khi người dùng di chuyển qua lại để chọn chế độ thì sẽ có những hình ảnh của mỗi chế độ khác nhau hiện ra.

d. Xử lý âm thanh/màu sắc

- **Âm thanh**

- Sử dụng `#pragma comment(lib, "winmm.lib")` để có thể phát được nhạc.
- Âm thanh trong trò chơi được chia làm hai loại: nhạc nền và âm thanh hiệu ứng.
- Có thể bật, tắt nhạc nền cũng như âm thanh hiệu ứng trong mục Option Setting của game hoặc trong mục Option trong lúc đang chơi.

- **Màu sắc**

- Sử dụng thư viện `Windows.h` để xử lý màu sắc.
- Một số hàm để điều chỉnh màu sắc như: `setTextColor`, `TextColor`, `FontSize`.
- Ngoài ra, đối với giao diện game hay các vật thể trong game cũng được vẽ lại để có thể thân thiện với người dùng hơn bằng cách tạo ra một mảng 2 chiều chứa kí tự rỗng “ ” và in ra theo màu sắc và vị trí được chọn.

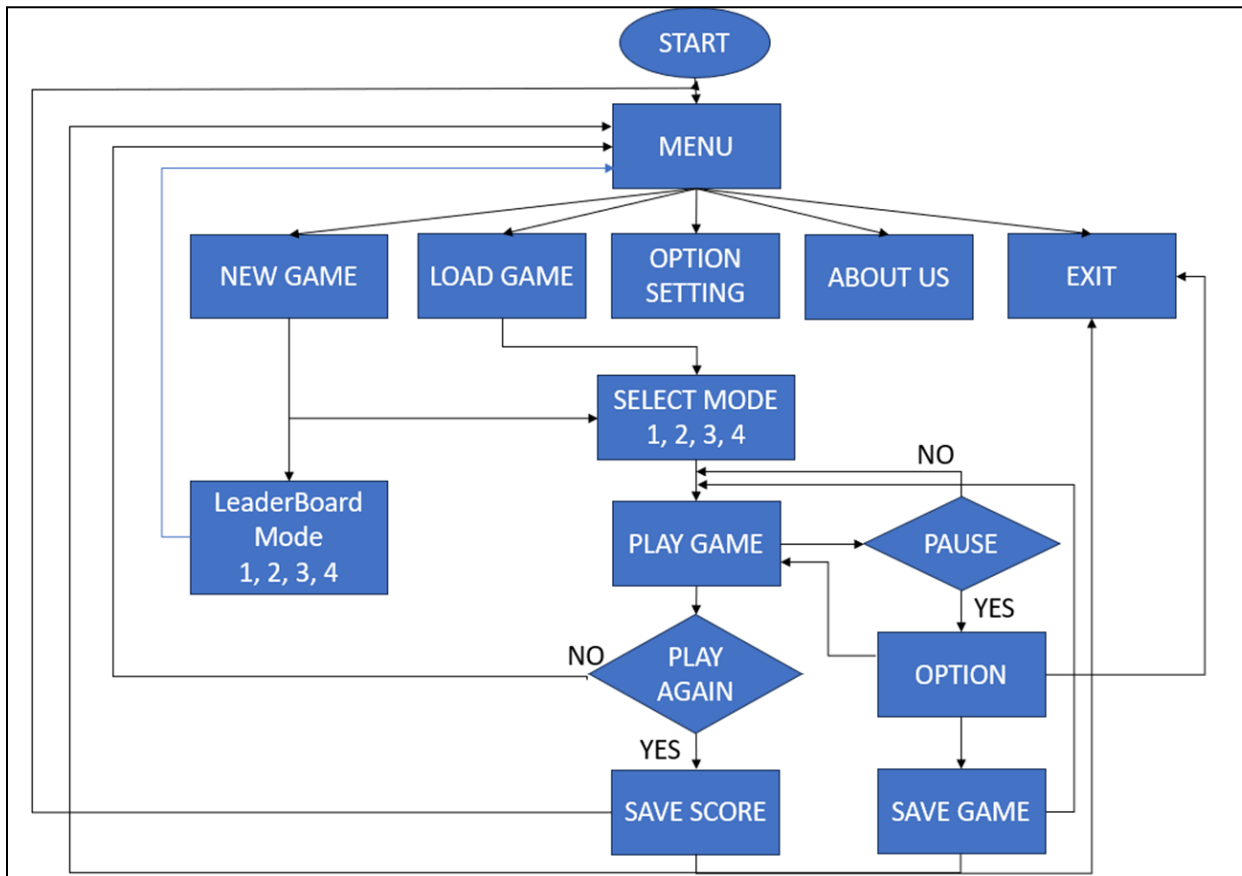
e. Tính năng bổ sung

- Một số tính năng được bổ sung:

- Có chế độ chơi cho 2 người.
- Thiết kế nhiều màn chơi ở chế độ 2 và chế độ 3.
- Có thể thay đổi bàn phím theo nhu cầu của người chơi.
- Có 5 loại quả khác nhau:
 - + Quả thường: rắn ăn được sẽ tăng 100 điểm.
 - + Quả đặc biệt: rắn ăn được sẽ tăng 500 điểm.
 - + Quả choáng: rắn ăn phải sẽ gây rối loạn di chuyển (khi rắn đi lên sẽ đi xuống, rắn qua phải sẽ qua trái và ngược lại).
 - + Quả độc: rắn ăn phải sẽ bị giảm đốt và bị trừ điểm.
 - + Quả tăng tốc: rắn ăn quả này sẽ tăng tốc độ di chuyển.
- Người chơi có thể thua cuộc khi chạm vào chướng ngại vật như vật cản, thành của cổng qua màn, chạm vào rắn của đối thủ hay ăn quá nhiều quả bị giảm đốt.

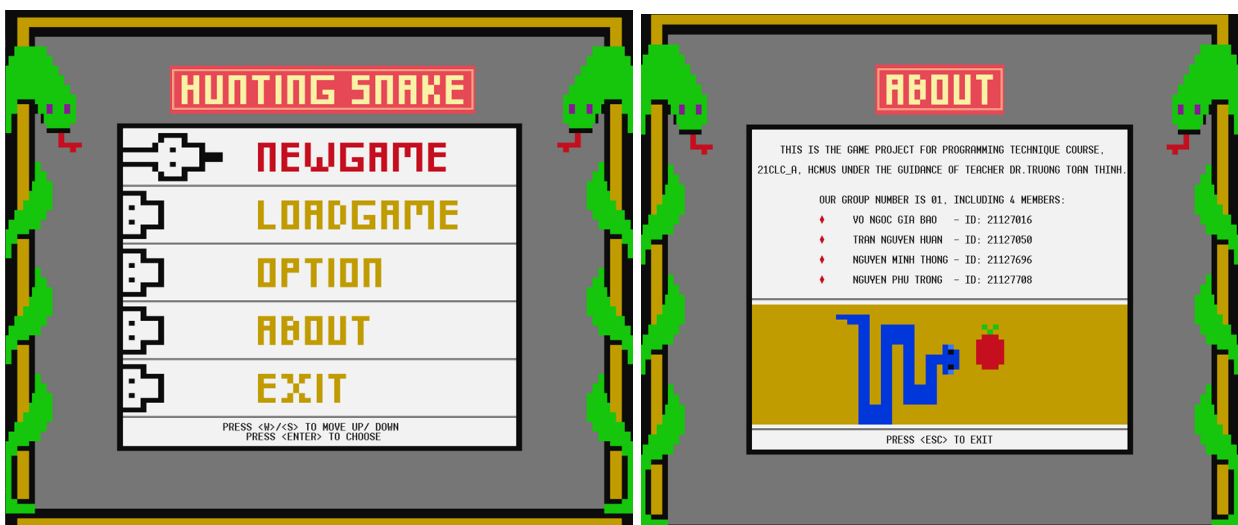
3. Các hình ảnh về trò chơi

a. Lưu đồ của trò chơi



Lưu đồ của trò chơi

b. Menu và các chế độ chơi

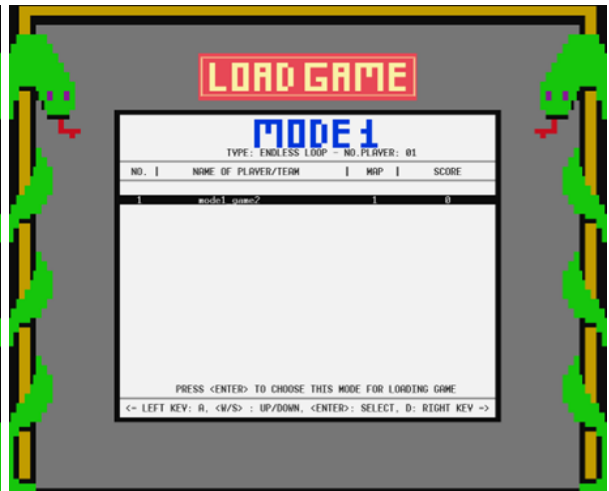


Menu chính

About us



Menu Save Game



Menu Load Game



Leaderboard



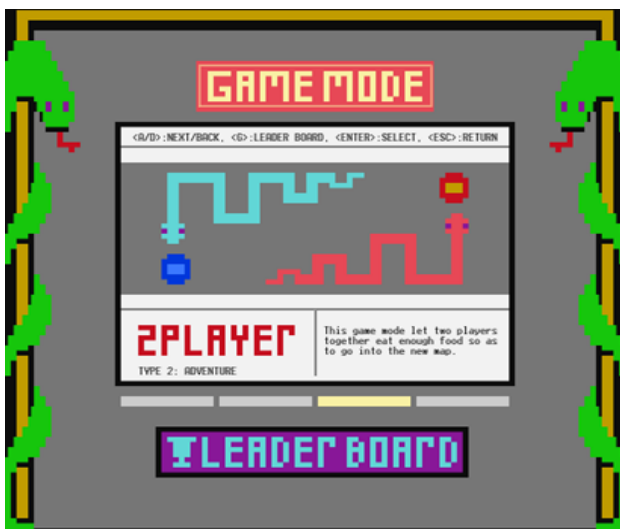
Guide



Mode 1: Chơi đơn - vô tận



Mode 2: Chơi đơn - phiêu lưu



Mode 3: Chơi đội - phiêu lưu



Mode 4: Đối kháng - 2 người

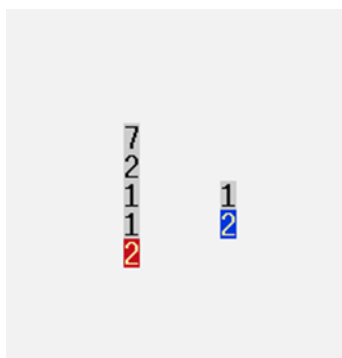


Thua Game



Thắng Game

c. Các yếu tố trong trò chơi



Rắn xanh và rắn đỏ



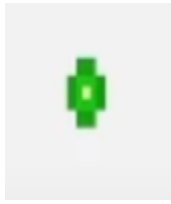
Quả đỏ



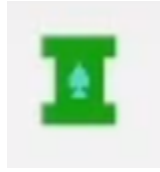
Quả xanh



Quả đặc biệt



Quả choáng



Quả độc



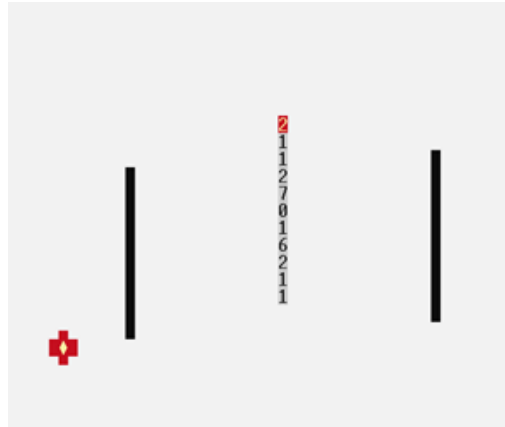
Quả tăng tốc



*Cổng qua màn
của rắn xanh*



*Cổng qua màn
của rắn đỏ*



Chương ngại vật, rắn và quả

PHẦN 3

TỔ CHỨC MÃ NGUỒN

1. Xây dựng các yếu tố chính trong game

a. Nhân vật (rắn)

❖ Thuộc tính chính của rắn bao gồm:

- **TenNguoiChoi** (tên người chơi): chuỗi char* lưu trữ tên của người chơi đặt sau khi đạt được số điểm sau khi thua hoặc thắng trò chơi.
- **BodyRan** (thân rắn): biến int là số nguyên chỉ đơn vị chiều dài mà rắn đã đạt được trong quá trình chạm các vật phẩm.
- **Tọa độ của rắn**: một đối tượng struct lưu lại thông tin tọa độ x, y của từng ô chỉ về thân của con rắn, giúp xác định vị trí đầu, thân và kiểm tra va chạm vật thể của rắn.

- **SPEED** (tốc độ): tốc độ của rắn là mức độ trì hoãn để in ra vị trí tiếp theo của rắn khi đang di chuyển, tốc độ càng thấp thì sự trì hoãn càng rút ngắn, rắn di chuyển nhanh hơn.
- **Màn**: biến int lưu lại con số màn chơi mà rắn đang ở.
- **Hướng**: biến int trong khoảng từ 1 đến 4, tức chỉ các hướng mà rắn sẽ di chuyển. Ngoài ra giúp điều hướng vị trí của rắn xét theo vị trí đầu của rắn hiện tại.
- Các yếu tố khác như vật cản, vật cản động, điểm ... đều liên quan đến quá trình xử lý trò chơi - nằm trong đối tượng Người chơi.
- ❖ Các hàm xử lý liên quan đến người chơi:
 - **KhoiTaoRan** (khởi tạo rắn): tạo rắn với vị trí tọa độ x, y.
 - **VeRan** (vẽ rắn), **HienThiRan** (hiển thị rắn) và **DiChuyen** (rắn di chuyển): giúp in hình dáng của rắn dựa theo tọa độ mà rắn đang ở.
 - **RanChamThan** (rắn chạm thân), **RanChamTuong** (rắn chạm tường), **RanChamCongWin** (rắn chạm công qua màn) và **RanChamNhuau** (rắn chạm nhau): là những hàm sẽ xét điều kiện thua game khi các yếu tố liên quan đến tọa độ của rắn có thỏa điều kiện nhất định.
 - **KiemTraAnQua** (kiểm tra ăn quả) và các hàm liên quan đến ăn quả: là nhóm hàm giúp kiểm tra xem tọa độ của phần tử đầu tiên của nhân vật - đầu rắn có chạm đến vị trí của các vật phẩm tăng điểm hay không, nếu có thì sẽ tính điểm cho người chơi.
 - Các **hàm bắt sự kiện**: là những hàm sẽ thu nhận các tín hiệu từ bàn phím để cho biết rắn được người chơi di chuyển theo hướng nào.
 - **KiemTraDieuKienThang** (kiểm tra điều kiện thắng): mỗi màn chơi sẽ có những mức điểm khác nhau, hàm sẽ kiểm tra điểm trong đối tượng người chơi có đạt đủ yêu cầu để khởi tạo công qua màn hay không.

b. Vật phẩm

- ❖ Thuộc tính chính của vật phẩm - chủ yếu là quả bao gồm:
 - **Count** (đếm số lượng): biến int chỉ ra số lượng mà vật phẩm đó đã xuất hiện ở trong màn chơi tính đến một thời điểm nhất định.
 - **Kiểm tra quả**: biến bool kiểm tra các biến thể trong vật phẩm quả đã đạt mức độ nào đó hay chưa để làm xuất hiện một biến thể đặc biệt hơn.
 - **Tọa độ quả thường**: đây là tọa độ x, y của quả giúp tăng 100 điểm trong game. Vì mức độ tăng điểm thấp nên gọi là quả thường.
 - **Tọa độ quả đặc biệt**: đây là tọa độ x, y của quả đặc biệt giúp tăng 500 điểm trong game.

- **Tọa độ quả hiệu ứng xấu:** đây là tọa độ x, y của quả gây ra hiệu ứng xấu cho người chơi, khiến cho người chơi có thể thua nhanh chóng.
- ❖ Các hàm xử lý liên quan đến vật phẩm quả:
 - **TaoQuaThuong** (tạo quả thường), **VeQuaThuong** (vẽ quả thường) và các hàm biến thể về quả thường: khởi tạo vị trí của quả thường, vẽ và hiển thị nó lên trên màn chơi.
 - **TaoQuaDacBiet** (tạo quả đặc biệt), **VeQuaThuong** (vẽ quả đặc biệt) và các hàm biến thể về quả đặc biệt: khởi tạo vị trí của quả đặc biệt, vẽ và hiển thị nó lên trên màn chơi.
 - **TaoQuaDoc** (tạo quả độc), **VeQuaDoc** (vẽ quả độc) và các hàm liên quan đến quả độc: thực hiện chức năng như trên nhưng đối với quả độc.
 - **XoaQua** (xóa quả): làm hàm làm biến mất các vật phẩm khi người chơi tương tác với chúng, ghi đè hình ảnh vật phẩm in ra được bằng các kí tự khoảng trống để che lấp.
 - **DiXuyenQua** (Đi xuyên quả) là hàm cho phép rắn đi xuyên qua quả nếu như rắn không thể tương tác được nếu như khác loại với nhau, áp dụng cho Mode 3 và 4.
 - **AnQuaThuong** hoặc gọi chung là hàm ăn quả: đặt trong mỗi file màn chơi, nó sẽ kết hợp hàm khởi tạo quả, vẽ quả và in nó lên trên màn hình theo điều kiện ràng buộc: không được xuất hiện trong phạm vi thân rắn, chướng ngại vật, tường. Đồng thời, kiểm tra va chạm của rắn với quả để xác định tăng điểm cho người chơi. Đây là hàm quan trọng của trò chơi.

c. Chương ngại vật

- ❖ Thuộc tính chính của chương ngại vật bao gồm: Tọa độ vật cản có thuộc tính lấy từ struct Tọa độ và biến int chỉ số nguyên là kích thước của vật dùng để xử lý các hành vi va chạm giữa rắn và vật cản. Các thông tin vừa rồi được đóng gói bên trong struct NhanVat.
- ❖ Các hàm xử lý liên quan đến chương ngại vật:
 - **Cổng Win:**
 - **TaoCongWin** và các hàm biến thể TaoCongWin (tạo cổng win - qua màn): là hàm khởi tạo cổng qua màn khi người chơi đạt số điểm quy định của từng màn.
 - **VeCongWin:** dùng để in hình dáng của cổng đó tại vị trí quy định sẵn với 2 hướng: dọc - ngang.

- **RanChamCongWin** (rắn chạm cổng win): là hàm sẽ kiểm tra rằng liệu rắn có chạm 2 bên thành của cổng hay không. Nếu có thì nó sẽ báo hiệu rằng game đã kết thúc, người chơi thua cuộc.
 - **RanQuaMan** (rắn qua màn): là hàm dùng để kiểm tra vị trí đầu của rắn có chạm đến điểm chính giữa của cổng hay không. Nếu có thì sẽ trả về tín hiệu màn đã vượt qua, người chơi đi đến màn tiếp theo.
- **Vật cản:**
- **TaoChuongNgaiVat** (tạo chương ngại vật): là hàm tạo ra chương ngại vật, nhận đầu vào là mảng tọa độ vật cản, số nguyên kích thước vật cản. Sau đó sẽ chọn từng tọa độ x, y trong bản đồ trò chơi để làm tọa độ của vật cản.
 - **KiemTraThuaVatCan** (kiểm tra chạm vật cản): là hàm kiểm tra liệu một trong các phần tử của thân rắn có chạm đến một trong các phần tử của chương ngại vật không. Nếu có sẽ trả về tín hiệu dừng trò chơi và người chơi thua cuộc.
- **Tường:** đây là đối tượng chương ngại vật được quy định sẵn trong quá trình tạo ra khung sườn bản đồ trò chơi. Các hàm xử lý giống vật cản trên.

d. Giao diện game

❖ Đây là nhóm các đối tượng bao gồm:

➤ **Bên ngoài trò chơi:**

- Giao diện menu chính (giao diện khi vào game).
- Giao diện chọn chế độ màn chơi.
- Giao diện load game: tải màn chơi đã lưu.
- Giao diện setting: đây là giao diện thiết lập thông số trước chơi.
- Giao diện about: hiển thị thông tin đồ án và nhóm.

➤ **Bên trong trò chơi:**

- Giao diện chơi game: bản đồ chơi, thanh điểm số và chức năng.
- Giao diện khi dừng trò chơi.
- Giao diện khi chọn mục Guide.
- Giao diện khi chọn mục Option trong game.
 - Giao diện setting trong mục option.
 - Giao diện save game: lưu màn chơi.

❖ Các hàm xử lý liên quan đến các giao diện trong game:

- **Vẽ Giao Diện:** đây là hàm sẽ vẽ đối tượng phụ họa như hình ảnh rắn 2 bên viền của khung. Các text box với nền và khung viền đã được tô màu.
- **Vẽ Chữ:** đây là hàm sử dụng các kí tự ghép nối lại với nhau được bao gồm trong một mảng ký tự 2 chiều. Mỗi ký tự lấy từ trong bảng mã ASCII.
- **Vẽ Số:** tương tự như trên, dùng để in các chữ số.
- **Vẽ Kí hiệu:** đây là hàm vẽ các đối tượng không phải là chữ cũng không phải là số. Giúp phụ họa cho chương trình.
- **Hàm xử lý giao diện:** đây làm sẽ xử lý các tín hiệu điều khiển từ người dùng. Khi người chơi có sự lựa chọn hoặc thay đổi, giao diện sẽ tương tác lại với người chơi:
 - Đầu tiên là sẽ hiển thị thay đổi các nút chọn như thay đổi màu sắc, thêm thắt kí tự, hình vẽ để cho thấy người chơi đang chọn option đó.
 - Sau đó, khi người chơi nhấn kí tự di chuyển lên xuống hoặc trái phải và nhấn [ENTER] thì chương trình sẽ trả về kết quả của sự lựa chọn đó cho các hàm sau xử lý để đi đến mục tiêu mà người dùng hướng đến. Hoặc nhấn [ESC] để trở về giao diện lúc đầu.

2. Tổ chức tập tin

Trong quá trình xây dựng trò chơi, nhóm em đã sử dụng 2 loại tập tin chính đó là tập tin âm thanh (.mp3 và .wav) dùng để chạy nhạc nền, hiệu ứng âm thanh trong game và tập tin file text (.txt) dùng để lưu trữ nội dung trò chơi khi người dùng muốn lưu game và tải game để chơi lại.

2.1 Tập tin lưu trữ trò chơi

Tập tin lưu trữ nội dung trò chơi gồm có 1 file quản lý và các file chứa nội dung trò chơi. File quản lý đó sẽ chứa các tên file mà chứa nội dung trong game. Khi chương trình chạy, nó sẽ nạp các tên file từ trong chính file quản lý đó để giúp chương trình xác định được nội dung của tên file người chơi muốn sử dụng để tải trò chơi (load game).

Các file nội dung game sẽ chứa lần lượt các thông tin quan trọng như: thông tin chế độ, chiều dài thân rắn, tọa độ các điểm trên thân của rắn và điểm người chơi. Các thông tin đó khi người dùng muốn tải game, sẽ được chương trình khởi tạo màn chơi với các thông tin đã được ghi nhận trước đó. Sau khi khởi tạo lại xong, người chơi sẽ tiếp tục màn chơi tiếp.

2.2 Tập tin âm thanh

Các tập tin âm thanh như file .mp3 và file .wav được đặt trong cùng thư mục mã nguồn của trò chơi. Để chạy được các file trên, bên trong mã nguồn đã thiết lập sẵn cài đặt `#pragma comment(lib, "winmm.lib")`.

3. Lưu ý khi chạy mã nguồn

Mã nguồn chương trình được tạo ra từ Microsoft Visual Studio phiên bản 2022. Có sử dụng các thư viện như chrono - dùng để đo đạc, tính toán thời gian nhằm để xử lý một số hàm nhất định trong game, thư viện <Windows.h> dùng để xử lý các chức năng cơ bản liên quan đến màn hình console, thư viện <thread> xử lý tiến trình, đa luồng trong game. Ngoài ra chỉ sử dụng các thư viện có sẵn được hỗ trợ trong Visual Studio. Chương trình chạy tốt trên Hệ điều hành Windows.

Các lỗi có thể phát sinh: sự khác biệt về màn hình Console có thể sản sinh ra các giao diện có thành phần bị lỗi ký tự, màu sắc gây ảnh hưởng đến người dùng trải nghiệm. Các lỗi khác có tỉ lệ nhỏ phát sinh và đang được xử lý.

Khi thực hiện đồ án môn học - đồ án game Hunting Snake của môn Kỹ thuật lập trình lớp 22CLC_A, nhóm em đã có cơ hội được thực hiện việc xây dựng và phát triển một trò chơi tuổi thơ của rất nhiều người nói chung chung và nhóm em nói riêng.

Việc xây dựng và phát triển trò chơi này đã giúp tụi em có cơ hội được ôn tập và củng cố các kiến thức trong môn học ôn tập lần này. Ngoài ra củng cố thêm kinh nghiệm làm việc nhóm, quản lý một đồ án - điều sẽ giúp ích rất nhiều cho chúng em trong quá trình làm việc sau này. Nhóm em cảm ơn các thầy đã tạo ra và hỗ trợ nhóm trong quá trình hoạt động đồ án môn học lần này.