

Báo cáo môn Lập Trình Giao Diện

Viết chương Trình GAME LOTO

Giáo Viên: Nguyễn thị mai trang

Nguyễn Đức Huy , Lê Trung Hậu| Lập trình giao diện | 13/4/2023

# Giới thiệu

Lô tô không chỉ là trò giải trí đơn thuần mà còn gợi lại trong lòng mỗi người nhiều ký ức vui vẻ. Đó là những lúc cả gia đình hay bạn bè cùng quây quần bên nhau, cười nói rộn ràng. Đó là những lúc câu hò, tiếng rao lô tô cất lên trong sự chờ đợi hồi hộp. Hay tiếng "kinh" vang lên đầy hào hứng của người may mắn chiến thắng... Có thể nói trò chơi lô tô đã gắn liền với tuổi thơ của bao người bao thế hệ người Việt.. Trong báo cáo này, chúng ta sẽ tìm hiểu về cách viết chương trình chơi Lô tô thông qua ngôn ngữ lập trình C# Windows Form .NET

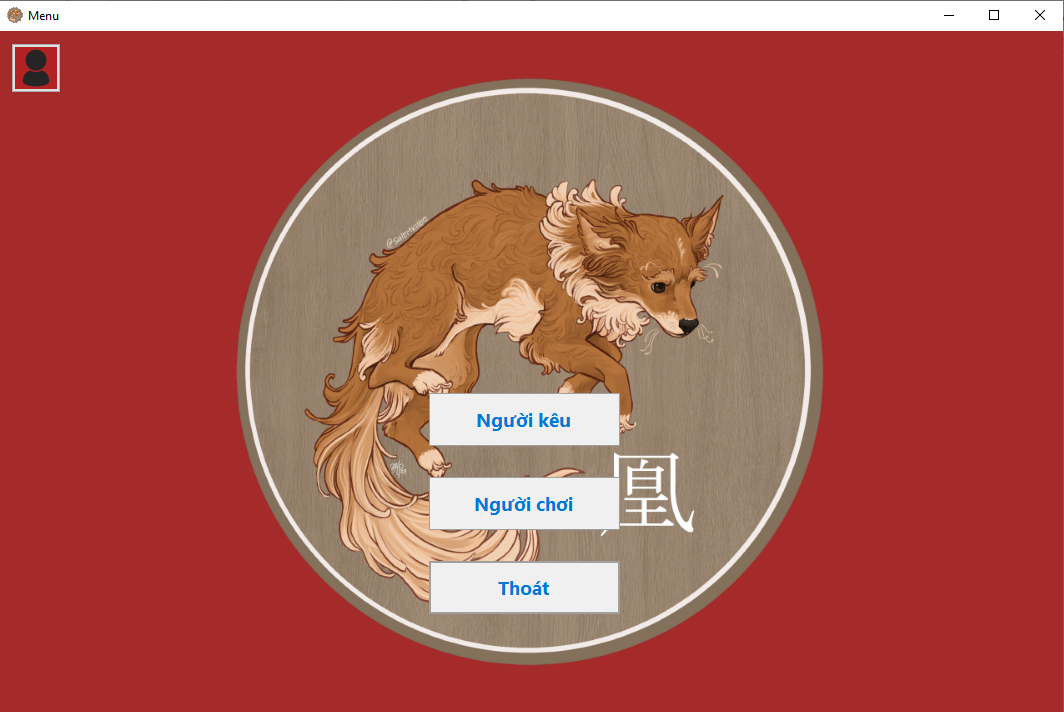
# Mô tả chương trình

* Chương trình sẽ kết nối 2 máy với nhau. Một máy đóng vai trò là máy Server, máy còn lại là máy Client
* Máy server sẽ tạo các con số Random từ 1 tới 90 gửi về cho Client.
* Bên phía Client là sẽ nhận được tính hiệu con số Random từ server gửi về để cho người dùng có thể kiễm tra và tích vào ô đã cố số
* Sau khi người dùng đã tích được 5 ô thẳng hàng ngang thì sẽ là người chiến thắng và được cộng thêm tiền.

# Các thành phần của chương trình

# Giao diện người dùng

Khi chương trình được chạy, sẽ hiện ra bảng để xem coi trong 2 người chơi thì người nào sẽ là máy Server (dùng để Random) tức là NGƯỜI KÊU , người còn lại sẽ là máy Client ( dùng để chơi ) tức là NGƯỜI CHƠI và có thêm chức năng thoát để THOÁT khỏi chương trình. Ở bên góc trái màn hình là sẽ là thông tin người chơi ( như TÊN, SỐ TIỀN , SỐ LẦN THẮNG và có thể thay đổi ẢNH ĐẠI DIỆN )





Thông tin người chơi

Sau khi bấm nút NGƯỜI KÊU, chương trình hiện ra các thông tin như Số PHÒNG, MENU CHÍNH. QUAY SỐ

Graphical user interface, application

Description automatically generated

Sau đó người chơi có thể chọn option Bốc số để Random một con số.

Graphical user interface, application

Description automatically generated

Bên phía Client thì sẽ yêu cầu chúng ta phải nhập vào số phòng để mới được vô phòng

Graphical user interface, application

Description automatically generated

Sau khi nhập xong ta sẽ bấm vào nút vào Phòng. Nếu nhập vào phòng đúng nó sẽ đưa ra thông báo “Kết nối thành công”.Ngược lại thì nó sẽ thông báo “Phòng không tồn tại”.

Nếu chọn Về trang chủ thỉ nó sẽ về Menu lúc đầu.

Graphical user interface, application, Teams

Description automatically generated

Khi không kết nối thành công

Graphical user interface, application, Teams

Description automatically generated

Table

Description automatically generated

Khi kết nối thành công

A picture containing diagram

Description automatically generated

Khi về Menu

# Các phương thức xử lí

**Yêu cầu của chương trình**

**1.Về phần Server/Client**

.Về phía Client: Ta sẽ viết một hàm dùng để xem đã kết nối với Server hay chưa

Text

Description automatically generated

Đầu tiên ta sẽ tạo phương thức IPEndPoint ipe để lấy địa chỉ IP và Port hiện tại. Sau đó ta sẽ tạo phương thức Socket mới để chỉ định loại IP , Socket mà ta muốn và Giao thức truyền tải. Tiếp theo ta sẽ bắt try catch vào để kiểm tra xem nó đã Connect thành công chưa. Nếu rồi thì trả về true, ngược lại trả về false

.Về phía Server: ta sẻ dùng một hàm void dùng để tạo Server.

Text

Description automatically generated

Đầu tiên ta sẽ tạo phương thức IPEndPoint ipe để lấy địa chỉ IP và Port hiện tại. Sau đó ta sẽ tạo phương thức Socket mới để chỉ định loại IP , Socket mà ta muốn và Giao thức truyền tải. Chương trình gọi phương thức Listen() của socket để lắng nghe kết nối từ client tới server với giới hạn đợi là 10 kết nối. Sau đó, một đối tượng Thread mới được khởi tạo và gán cho biến acceptClient. Thread này sẽ chạy một hàm không có tham số, trong đó gọi phương thức Accept() của socket. Phương thức Accept() sẽ chặn luồng hiện tại cho đến khi có một kết nối từ client được chấp nhận. Khi đó, socket server sẽ trả về một socket mới để giao tiếp với client đó. Biến IsBackground của đối tượng Thread acceptClient được gán giá trị true để đảm bảo rằng nó sẽ được kết thúc khi chương trình chính kết thúc. Cuối cùng, Thread acceptClient được khởi động bằng cách gọi phương thức Start(). Khi có một kết nối từ client được chấp nhận, socket client sẽ được gán vào biến client để sử dụng cho các thao tác giao tiếp với client đó.

.Về phần BOTH: ta sẽ tạo các hàm sau:

Text

Description automatically generated

Text

Description automatically generated

Hàm Send(object data): Hàm này nhận vào một đối tượng (object) cần gửi đi, nén đối tượng đó thành một mảng byte bằng phương thức SerializeData, và gọi hàm SendData để gửi mảng byte đã nén đó tới socket client. Hàm trả về true nếu dữ liệu được gửi đi thành công, và ngược lại.

Hàm Receive(): Hàm này sẽ nhận dữ liệu từ socket client, lưu dữ liệu vào một mảng byte đã khởi tạo trước đó, sử dụng hàm ReceiveData để nhận dữ liệu từ socket client và sau đó giải nén dữ liệu đó thành đối tượng gốc bằng phương thức DeserializeData. Hàm trả về đối tượng gốc đã giải nén.

Hàm SendData(Socket target, byte[] data): Hàm này nhận vào đối tượng Socket và mảng byte cần gửi đi. Hàm sử dụng phương thức Send của Socket để gửi dữ liệu đó đi và trả về true nếu dữ liệu được gửi thành công.

Hàm ReceiveData(Socket target, byte[] data): Hàm này nhận vào đối tượng Socket và mảng byte để lưu trữ dữ liệu nhận được. Hàm sử dụng phương thức Receive của Socket để nhận dữ liệu và trả về true nếu nhận được ít nhất một byte dữ liệu.

Hàm SerializeData(Object o): Hàm này nhận vào một đối tượng (object) cần nén, tạo ra một đối tượng MemoryStream để lưu trữ dữ liệu nén, sử dụng BinaryFormatter để nén đối tượng vào MemoryStream, và trả về mảng byte của MemoryStream.

Hàm DeserializeData(byte[] theByteArray): Hàm này nhận vào một mảng byte đã nén, tạo ra một đối tượng MemoryStream để lưu trữ dữ liệu giải nén, sử dụng BinaryFormatter để giải nén dữ liệu từ mảng byte vào MemoryStream, và trả về đối tượng gốc đã giải nén.

Hàm GetLocalIPv4(NetworkInterfaceType \_type): Hàm này nhận vào một tham số kiểu NetworkInterfaceType để chỉ định loại card mạng cần lấy địa chỉ IP, duyệt qua tất cả các card mạng trên máy tính và trả về địa chỉ IP của card mạng đó nếu nó đang được sử dụng. Hàm trả về một chuỗi rỗng nếu không tìm thấy địa chỉ IP phù hợp.

**2.Về phần Sinh số ngẫu nhiên**

Ta sẽ dùng một hàm RandomNumber để t sinh các số ngẫu nhiên từ 1 tới 90

Text

Description automatically generated

Ta sẽ dùng một List tên là result để lưu các kết quả đã sinh ra sau đó sẽ add vào hàm Appear để nó được xuất hiện trên máy Server. Nếu số đó đã được sinh ra từ trước thì ta sẽ Random lại số đó.

Text

Description automatically generated

**3.Về phần tạo giấy và chọn giấy mới khi chơi xong**

Ta sẽ tạo hàm CreatPaper() để tạo các tờ giấy Lô tô

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Đầu tiên ta sẽ ghi số tiền mà trò chơi mặc định cung cấp khi chơi. Ta sẽ dùng vặp for để duyệt các cột. Dòng for đầu tiên là tạo 8 hàng với mỗi hàng có 9 con số từ 1 tới 9 hoặc 11 đến 19, 21 đến 29, … , được sắp xếp ngẫu nhiên. Dòng for thứ 2 dùng để tạo hàng thứ 9 với 9 số ngẫu nhiên từ 80 đến 90 và một số trống ngẫu nhiên. Dòng for thứ 3 tạo 4 ô trống ngẫu nhiên trên mỗi hàng và một ô trống ngẫu nhiên trong các ô trống đó sẽ chứa số trống. Sau đó ta sẽ xóa kết quả trước đó nếu có.

Khi chúng ta chơi xong chúng ta có thể đổi tờ giấy mới bằng cách xóa tờ giấy cũ và thực hiện lại bước tạo giấy như trên

Text

Description automatically generated

**3.Về phần tạo giấy và chọn giấy mới khi chơi xong**

Ta sẽ tạo hàm bool dùng để kiểm tra ta đã thắng hay chưa.Text

Description automatically generated

Hàm này sẽ kiểm tra xem nếu đã đủ con số ở hàng ngang thì nó sẽ trả về true còn không thì sẽ trả về false.

Sau khi trả về true tức là đã thắng thì ta sẽ cộng tiền cho nó và sau khi thua thì sẽ trừ tiền và cập nhật lại tiền khi ta cộng hoặc trừ . Nếu mà không còn đồng nào thì ta sẽ ra thông báo "Bạn đã cháy túi >.<!" \

Text

Description automatically generated

**---HẾT---**