**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**---------------------------------------**



**BÀI TẬP LỚN**

**Môn: An toàn và bảo mật thông tin**

**Giấu tin trong ảnh sử dụng kết hợp mã hóa AES và kỹ thuật giấu tin trên sai phân.**

|  |
| --- |
| **GVHD: *ThS. Trần Phương Nhung*** |
| **Nhóm: 11** |
| **Thành viên nhóm:**   1. **Hoàng Anh Nghiệp – 2019605727** 2. **Mạch Văn Quân – 2019600865** 3. **Dương Hải Nguyên – 2019600082** 4. **Phạm Văn Nhất – 2019600845** 5. **Đinh Thị Hồng Nhung – 2019602279** |

Hà Nội – Năm 2022

**Mục lục**

[Chương 1: Tổng quan 4](#_Toc113552581)

[1.1. Mục đích chọn đề tài 4](#_Toc113552582)

[1.2. Nội dung nghiên cứu 4](#_Toc113552583)

[1.3. Các kiến thức cần có 4](#_Toc113552584)

[1.3.1. Các khái niệm 4](#_Toc113552585)

[1.3.2. Tính chất của mã hóa thông tin 5](#_Toc113552586)

[1.3.3. Độ an toàn của hệ mã 5](#_Toc113552587)

[1.3.4. Các phương pháp mã hóa 5](#_Toc113552588)

[1.3.5. Ứng dụng của mã hóa thông tin 6](#_Toc113552589)

[1.3.6. Giới thiệu một số giải thuật mã hóa tiên tiến 6](#_Toc113552590)

[1.4. Lĩnh vực hoạt động của đề tài 7](#_Toc113552591)

[Chương 2: Kết quả nghiên cứu 8](#_Toc113552592)

[2.1. Giới thiệu 8](#_Toc113552593)

[2.2. Nội dung thuật toán mã hóa AES 9](#_Toc113552594)

[2.2.1. Giới thiệu 9](#_Toc113552595)

[2.2.2. Quy trình mã hóa 10](#_Toc113552596)

[2.2.3. Quy trình giải mã 11](#_Toc113552597)

[2.3. Thiết kế giao diện demo 12](#_Toc113552598)

[2.4. Cài đặt và triển khai 13](#_Toc113552599)

[2.4.1. Giới thiệu về công cụ 13](#_Toc113552600)

[2.4.2. Cài đặt và chạy chương trình demo 15](#_Toc113552601)

[2.5. Thực hiện bài toán 22](#_Toc113552602)

[2.5.1. Phân công công việc 22](#_Toc113552603)

[2.5.2. Hoàng Anh Nghiệp – Phương pháp đánh giá ảnh và ngôn ngữ C++ 22](#_Toc113552604)

[2.5.3. Mạch Văn Quân – Tìm hiểu về mật mã AES và ngôn ngữ Ruby 29](#_Toc113552605)

[2.5.4. Phạm Văn Nhất – Tìm hiểu giấu tin và ngôn ngữ Java 34](#_Toc113552606)

[2.5.5. Dương Hải Nguyên – Tìm hiểu cấu trúc ảnh Bitmap và ngôn ngữ C# 39](#_Toc113552607)

[2.5.6. Đinh Thị Hồng Nhung – Tìm hiểu giấu tin trên sai phân và ngôn ngữ Python 47](#_Toc113552608)

[Chương 3: Kiến thức lĩnh hội và bài học kinh nghiệm 55](#_Toc113552609)

[3.1. Nội dung đã thực hiện 55](#_Toc113552610)

[3.2. Hướng phát triển 57](#_Toc113552611)

# Chương 1: Tổng quan

## Mục đích chọn đề tài

Hiện nay trong sự phát triển của thời đại kỷ nguyên số, thì công nghệ đóng vai trò không thể thiếu, đặc biệt là công nghệ bảo mật thông tin. Trong các ngân hàng hay trong các giao dịch điện tử việc bảo mật thông tin cá nhân của người dùng là vô cùng quan trọng. Chính vì như vậy mà trong bài tập lớn môn ATBMTT , chúng em - nhóm 11 đã chọn đề tài nghiên cứu kỹ thuật giấu tin trong ảnh kết hợp với hệ mã hóa AES và kỹ thuật giấu tin trên sai phân. Đề tài này chúng ta cần nắm được cách mã hóa thông tin bằng hệ mã hóa AES và cách giấu tin trong ảnh và kỹ thuật giấu tin trên sai phân.

## Nội dung nghiên cứu

* Tìm hiểu mật mã AES.
* Tìm hiểu về giấu tin trong ảnh và cấu trúc ảnh bitmap.
* Nghiên cứu kỹ thuật giấu tin trên sai phân.
* Tìm hiểu ngôn ngữ lập trình Python, Java, C#, PHP.

## Các kiến thức cần có

### Các khái niệm

1. *Mật mã học*

Mật mã học là một ngành khoa học nghiên cứu về việc giấu thông tin. Cụ thể hơn, mật mã học là ngành học nghiên cứu về những cách chuyển đổi thông tin từ dạng "có thể hiểu được" thành dạng "không thể hiểu được" và ngược lại.

Một số khái niệm trong mật mã học gồm: Mã hóa (encrypt hay encipher), Giải mã (Decrypt hay decipher), Bản rõ (Plaintext), Cipher (hay cypher), Khóa (Key).

1. *Hệ mật mã (Crypto System)*

Một hệ mật mã là bộ 5 (P,C,K,E,D) thõa mãn các tính chất sau:

1. P là không gian bản rõ: là tập hữu hạn các bản rõ có thể có.
2. C là không gian bản mã: là tập hữu hạn các bản mã có thể có.
3. K là không gian khóa: là tập hữu hạn các khóa có thể có.
4. Đối với mỗi k∈K, có một quy tắc mã hóa ek∈E và một quy tắc giải mã tương ứng dk∈D. Với mỗi ek: P → C và dk: C → P là những hàm mà dk(ek(x))=x cho mọi bản rõ x∈P. Hàm giải mã dk chính là ánh xạ ngược của hàm mã hóa ek.
5. *Nguyên tắc Kerckhoffs*

Một hệ mật mã sẽ được an toàn ngay cả khi tất cả mọi thứ trên hệ thống đó là công khai ngoại trừ khóa (key).

"Thuật toán mã hóa được tạo ra không cần phải giữ bí mật, có thể được công bố công khai, rơi vào tay quân địch mà không có bất kỳ sự phiền phức nào cả".

### Tính chất của mã hóa thông tin

Mã hóa thông tin phải đảm bảo các tính chất sau: Tính bí mật (Confidentiality), tính xác thực (Authentication), tính toàn vẹn (Integrity).

### Độ an toàn của hệ mã

Độ an toàn của thuật toán phụ thuộc vào độ phức tạp của nó. Các yếu tố xem xét thuật toán an toàn là chi phí hay phí tổn, thời gian cần thiết để phá vỡ, lượng dữ liệu để phá vỡ.

### Các phương pháp mã hóa

* *Mã hóa cổ điển (Classical cryptography)*

Phương pháp này là tiền thân của các phương pháp mã hóa đối xứng ngày nay. Có hai phương pháp nổi bật đó là: Mã hoá thay thế (Substitution Cipher), Mã hoá hoán vị (Transposition Cipher)

* *Mã hoá đối xứng (Symetric cryptography)*

Mã hoá đối xứng sử dụng cùng một khoá cho cả hai quá trình mã hoá và giải mã. Mã hoá đối xứng có thể tác động trên bản rõ theo từng nhóm bit hay theo từng bit một.

* *Mã hoá bất đối xứng (Asymetric cryptography)*

Mã hóa bất đối xứng được thiết kế sao cho khoá sử dụng trong quá trình mã hoá khác biệt với khoá được sử dụng trong quá trình giải mã. Tất nhiên không thể suy luận khóa giải mã từ khóa mã và ngược lại. Khoá để mã hoá được gọi là khóa công khai (Public Key), khoá để giải mã được gọi là khóa bí mật (Private Key).

* *Hệ thống mã hoá khoá lai (Hybrid Cryptosystems)*

Hệ thống mã hoá khoá lai ra đời là sự kết hợp giữa tốc độ và tính an toàn của hai hệ thống mã hoá ở trên.

### Ứng dụng của mã hóa thông tin

Mã hóa thông tin được ứng dụng trong rất nhiều lĩnh vực cả về phần cứng và phần mềm.

### Giới thiệu một số giải thuật mã hóa tiên tiến

* *Các hệ mã khối*

Các hệ mã khối dựa trên cơ sở làm việc với các khối dữ liệu là các chuỗi bit có kích thước khau nhau (tối thiểu là 64bit), khóa của hệ mã hóa cũng là một xâu bit có độ dài cố định.

Một số giải thuật được sử dụng khá phổ biến là DES, Triple DES (3DES),

AES.

* Mã hóa DES

DES (Data Encryption Standard) là thuật toán mã hóa với dữ liệu đầu vào và đầu ra là một khối 64 bit với độ dài khóa 64 bit (trong đó 8 bit được dùng để kiểm tra tính chẵn lẻ). Thuật toán thực hiện 16 vòng với 16 khóa (48bit) được sinh ra trong mỗi vòng.

Quá trình giải mã được diễn ra tương tự nhưng với các khóa con ứng dụng vào các vòng trong theo thứ tự ngược lại.

Thuật toán DES bộc lộ một số điểm yếu mà những kẻ lợi dụng nó để thám mã.

* Triple DES (3DES)

Triple DES thật chất là mã hóa theo DES ba lần với khóa K1, K2, K3 cho mỗi

lần.

* Chuẩn mã hóa nâng cao AES

AES (Advanced Encryption Standard) là một chuẩn mã hóa cao cấp với khóa bí mật cho phép xử lý các khối dữ liệu đầu vào có kích thước 128 bit và sử dụng các khóa có độ dài 128, 192, 256 bit.

* *Các hệ mã hóa công khai*

Mã hoá bằng khoá công khai là phương thức được thực hiện trên hai khóa, một được dùng để mã hóa (được gọi là khóa công khai – public key) và một khóa được dùng trong quá trình giải mã (gọi là khóa bí mật – private key). Khóa giải mã không thể tính toán được từ khóa mã hóa.

* *Hàm băm*

Hàm băm là hàm toán học chuyển đổi một thông điệp có độ dài bất kỳ thành một dãy bit có độ dài cố định. Mọi thay đổi dù là rất nhỏ trên thông điệp đầu vào đều làm thay đổi giá trị băm của nó.

## Lĩnh vực hoạt động của đề tài

Ngày nay, khi ảnh số đã được sử dụng rất phổ biến, thì giấu thông tin trong ảnh đã đem lại rất nhiều những ứng dụng quan trọng trên nhiều lĩnh vực trong đời sống xã hội.

Ví dụ như đối với các nước phát triển, chữ kí tay đã được số hoá và lưu trữ sử dụng như là hồ sơ cá nhân của các dịch vụ ngân hàng và tài chính, nó được dùng để xác thực trong các thẻ tín dụng của người tiêu dùng. Phần mềm WinWord của MicroSoft cũng cho phép người dùng lưu trữ chữ kí trong ảnh nhị phân rồi gắn vào vị trí nào đó trong file văn bản để đảm bảo tính an toàn của thông tin. Tài liệu sau đó được truyền trực tiếp qua máy fax hoặc lưu truyền trên mạng.

Theo đó, việc nhận thực chữ kí, xác thực thông tin đã trở thành một vấn đề cực kì quan trọng khi mà việc ăn cắp thông tin hay xuyên tạc thông tin bởi các tin tặc đang trở thành một vấn nạn đối với bất kì quốc gia nào, tổ chức nào. Thêm vào đó, lại có rất nhiều loại thông tin quan trọng cần được bảo mật như những thông tin về an ninh, thông tin về bảo hiểm hay các thông tin về tài chính, các thông tin này được số hoá và lưu trữ trong hệ thống máy tính hay trên mạng.

Chúng rất dễ bị lấy cắp và bị thay đổi bởi các phần mềm chuyên dụng. Việc nhận thực cũng như phát hiện thông tin xuyên tạc đã trở nên vô cùng quan trọng, cấp thiết.

# Chương 2: Kết quả nghiên cứu

**Những nhiệm vụ, công việc chính khi thực hiện đề tài cùng kết quả đạt được:**

1. **Nội dung công việc**

* Tìm hiểu mật mã AES.
* Tìm hiểu về giấu tin trong ảnh và cấu trúc ảnh bitmap.
* Nghiên cứu kỹ thuật giấu tin trên sai phân.
* Tìm hiểu ngôn ngữ lập trình Python, Java, C#, C++,…
* Ứng dụng xây dựng chương trình giấu tin trong ảnh sử dụng kết hợp mã hóa AES và kỹ thuật giấu tin trên sai phân.
* Demo chương trình.

1. **Kết quả đạt được**

* Đọc tài liệu và hiểu được vấn đề đặt ra, nắm được các phương pháp mã hóa AES một cách thành thạo (cả tiếng việt và tiếng anh).
* Hiểu được kỹ thuật giấu tin trên sai phân.
* Đọc hiểu được một số tài liệu chuyên môn bằng tiếng Anh.
* Nắm vững ngôn ngữ lập trình Python, Java, C#, C++ và giải được bài toán có tính ứng dụng vào thực tiễn.

## Giới thiệu

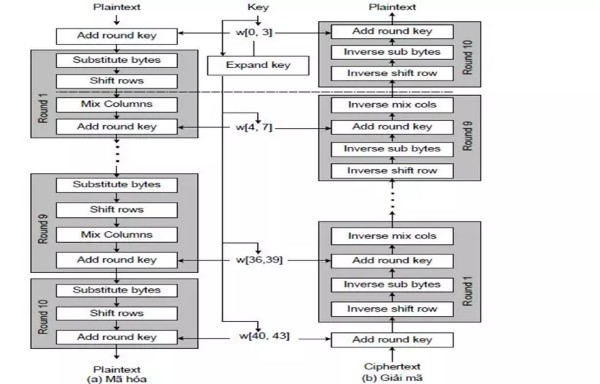
* **Tên đề tài**: Giấu tin trong ảnh sử dụng kết hợp mã hóa AES và kỹ thuật giấu tin trên sai phân.
* **Các bước thực hiện triển khai đề tài**:
* Nghiên cứu nội dung của các thuật toán mã hóa AES, thuật toán giấu tin trong ảnh.
* Thiết kế và cài đặt chương trình demo các thuật toán trên.
* **Hình thức sản phẩm**: Sản phẩm là một chương trình ứng dụng triển khai các thuật toán mã hóa AES và thuật toán giấu tin trong ảnh.
* **Kết quả đạt được**:
* Hiểu được cách thức hoạt động của thuật toán mã hóa AES và thuật toán giấu tin trong ảnh.
* Chương trình demo thực hiện được các chức năng cơ bản về mã hóa, giải mã và giấu tin trong ảnh.

## Nội dung thuật toán mã hóa AES

### Giới thiệu

Advanced Encryption Standard là thuật toán của hai nhà nghiên cứu Tiến sĩ Joan Daemon và Tiến sĩ Vincent Rijmen từ Bỉ.

Chuẩn mã hóa AES cho phép xử lý các khối dữ liệu đầu vào có kích thước 128 bit sử dụng các khóa có độ dài 128, 192 hoặc 256 bit. Thuật toán AES là một thuật toán khóa đối xứng có nghĩa là phím tương tự được sử dụng để mã hóa và giải mã tin nhắn. Ngoài ra, các thuật toán mã hóa văn bản được sản xuất bằng các thuật toán AES là như nhau kích thước như tin nhắn văn bản đơn giản. Hầu hết các hoạt động trong thuật toán AES xảy ra trên các byte dữ liệu hoặc trên từ dữ liệu dài 4 byte, được đại diện trong các trường GF (28), Được gọi là trường Galois. AES dựa trên một nguyên tắc thiết kế được biết đến như là một thay thế hoán vị mạng. AES hoạt động trên một ma trận 4 × 4 của byte, gọi là mảng trạng thái. Thuật toán mã hóa AES được quy định như một số lặp đi lặp lại vòng chuyển đổi đầu vào của bản rõ, thành quả cuối cùng sẽ là bản mã. Mỗi vòng bao gồm một số bước xử lý, trong đó có một phụ thuộc vào khóa mã hóa. Một tập hợp các vòng đảo ngược được áp dụng để biến đổi bản mã trở lại bản gốc bản rõ bằng cách sử dụng cùng một khóa mã hóa. AES thuật toán vòng lặp thông qua các phần nhất định Nr lần.



### Quy trình mã hóa

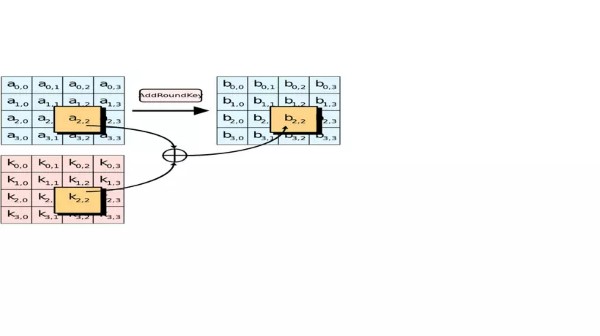
Bắt đầu quá trình mã hóa, bản rõ được sao chép vào mảng trạng thái. Sau khi thực hiện thao tác cộng với khóa mã đầu tiên, mảng trạng thái sẽ được biến đổi qua Nr vòng trong đó lần cuối cùng được thực hiện khác với Nr-1 vòng trước đó. Nội dung của mảng trạng thái ở vòng cuối cùng sẽ là bản mã của quá trình mã hóa.

Trong quy trình mã hóa của AES, tất cả các vòng lặp đều sử dụng 4 hàm theo thứ tự: Subbytes(), ShiftRows(), MixColumns(), AddRoundKey(). Riêng vòng cuối cùng bỏ qua việc gọi hàm MixColumns().

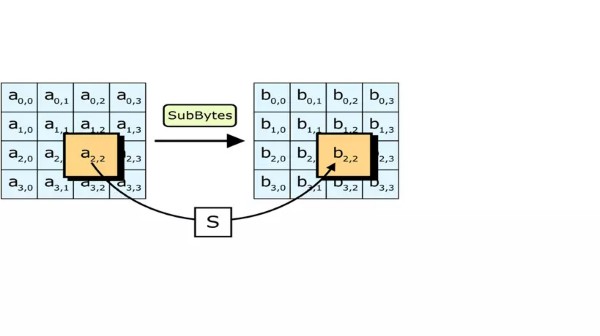
Quá trình mã hóa AES có các bước sau đây:

1. KeyExpansion -Round (khóa vòng mở rộng) được tạo ra từ khóa mã hóa bằng cách sử dụng lược đồ khóa Rijndael.
2. Initial Round (vòng khởi tạo)

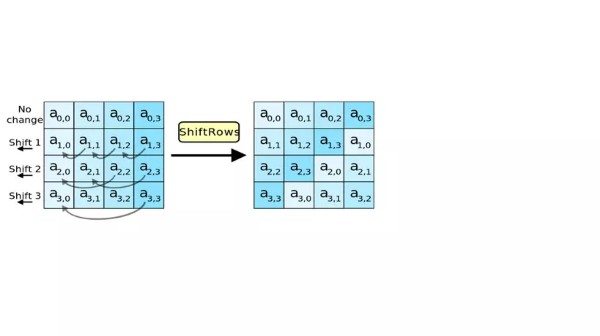
AddRoundKey - thực hiện bằng cách cộng một khóa vòng tại vòng đang xét với mảng trạng thái thông qua phép toán XOR đơn giản trên bit.



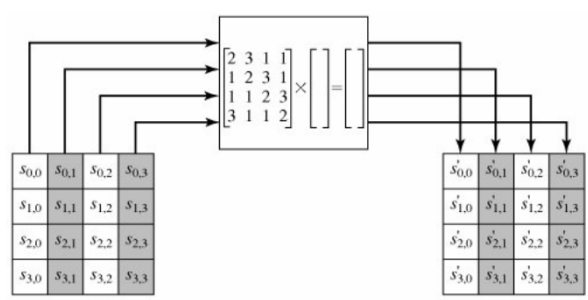
1. Rounds (vòng lặp)
2. SubBytes - làm biến mảng trạng thái hiện hành bằng cách sử dụng một bảng thay thế.



1. ShiftRows - làm biến đổi các byte trên ba hàng cuối cùng mảng trạng thái bằng cách dịch vòng.



1. MixColumns - là một phép biến đổi mã hóa được thực hiện bằng cách lấy tất cả các cột của mảng trạng thái trộn với dữ liệu của chúng một cách độc lập nhau để tạo ra các cột mới.



1. Final Round – vòng kết thúc (không có MixColumns)

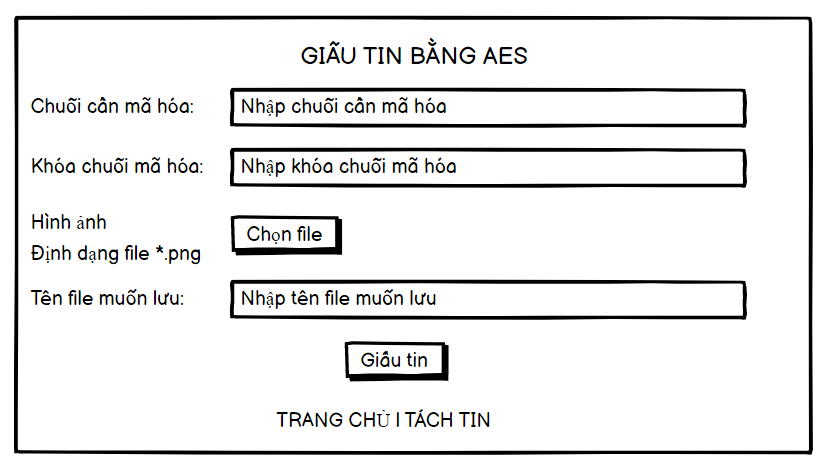
Gọi lại các hàm SubBytes, ShiftRows, MixColumns ở bước 3.

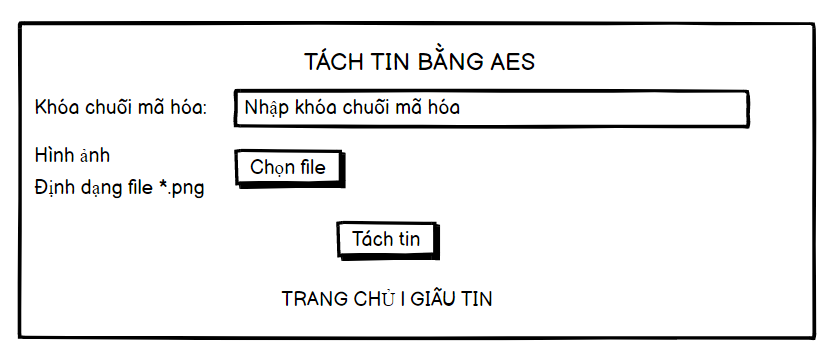
### Quy trình giải mã

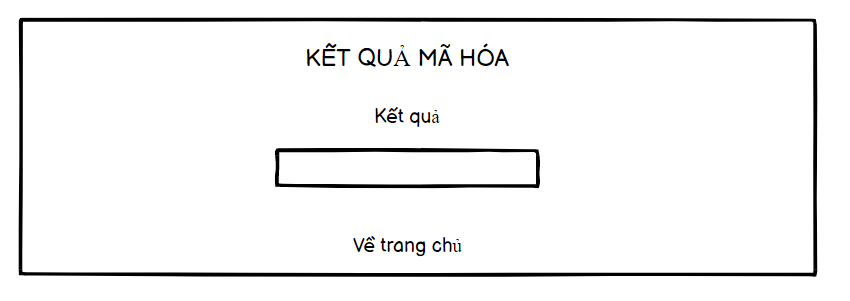
Quá trình giải mã được thực hiện theo chiều ngược lại với quy trình mã hóa, đồng thời các phép biến đổi trong quá trình này cũng được thực hiện đảo ngược. Ngoại trừ phép biến đổi AddRoundKey() không thay đổi vì chính bản thân nó là một phép biến đổi thuận ngịch do chỉ áp dụng một phép toán XOR.

* InvShiftRows() chính là phép biến đổi ngược của ShiftRows().
* Là phép biến đổi ngược của SubBytes() được thực hiện trên bảng thay thế S-Box là nghịch đảo của S-Box.
* InvMixColumns() là phép biến đổi ngược của MixColumns().

## Thiết kế giao diện demo







## Cài đặt và triển khai

### Giới thiệu về công cụ

* **Visual Studio**

Microsoft Visual Studio là một [môi trường phát triển tích hợp](https://vi.wikipedia.org/wiki/M%C3%B4i_tr%C6%B0%E1%BB%9Dng_ph%C3%A1t_tri%E1%BB%83n_t%C3%ADch_h%E1%BB%A3p) (IDE) từ [Microsoft](https://vi.wikipedia.org/wiki/Microsoft). Nó được sử dụng để phát triển [chương trình máy tính](https://vi.wikipedia.org/wiki/Ch%C6%B0%C6%A1ng_tr%C3%ACnh_m%C3%A1y_t%C3%ADnh) cho Microsoft Windows, cũng như các trang web, các ứng dụng web và các dịch vụ web. Visual Studio sử dụng nền tảng phát triển phần mềm của Microsoft như [Windows API](https://vi.wikipedia.org/wiki/Windows_API), [Windows Forms](https://vi.wikipedia.org/wiki/Windows_Forms), [Windows Presentation Foundation](https://vi.wikipedia.org/wiki/Windows_Presentation_Foundation), [Windows Store](https://vi.wikipedia.org/wiki/Windows_Store) và [Microsoft Silverlight](https://vi.wikipedia.org/wiki/Microsoft_Silverlight). Nó có thể sản xuất cả hai [ngôn ngữ máy](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_m%C3%A1y) và [mã số quản lý](https://vi.wikipedia.org/w/index.php?title=M%C3%A3_s%E1%BB%91_qu%E1%BA%A3n_l%C3%BD&action=edit&redlink=1).

Visual Studio bao gồm một [trình soạn thảo mã](https://vi.wikipedia.org/w/index.php?title=Tr%C3%ACnh_so%E1%BA%A1n_th%E1%BA%A3o_m%C3%A3&action=edit&redlink=1) hỗ trợ [IntelliSense](https://vi.wikipedia.org/w/index.php?title=IntelliSense&action=edit&redlink=1) cũng như [cải tiến mã nguồn](https://vi.wikipedia.org/wiki/C%E1%BA%A3i_ti%E1%BA%BFn_m%C3%A3_ngu%E1%BB%93n). Trình gỡ lỗi tích hợp hoạt động cả về trình gỡ lỗi mức độ mã nguồn và gỡ lỗi mức độ máy. Công cụ tích hợp khác bao gồm một mẫu thiết kế các hình thức xây dựng [giao diện ứng dụng](https://vi.wikipedia.org/w/index.php?title=Giao_di%E1%BB%87n_%E1%BB%A9ng_d%E1%BB%A5ng&action=edit&redlink=1), [thiết kế web](https://vi.wikipedia.org/wiki/Thi%E1%BA%BFt_k%E1%BA%BF_web), thiết kế [lớp](https://vi.wikipedia.org/wiki/L%E1%BB%9Bp_(khoa_h%E1%BB%8Dc_m%C3%A1y_t%C3%ADnh)) và thiết kế [giản đồ cơ sở dữ liệu](https://vi.wikipedia.org/w/index.php?title=Gi%E1%BA%A3n_%C4%91%E1%BB%93_c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u&action=edit&redlink=1). Nó chấp nhận các plug-in nâng cao các chức năng ở hầu hết các cấp bao gồm thêm hỗ trợ cho các [hệ thống quản lý phiên bản](https://vi.wikipedia.org/wiki/H%E1%BB%87_th%E1%BB%91ng_qu%E1%BA%A3n_l%C3%BD_phi%C3%AAn_b%E1%BA%A3n) (như [Subversion](https://vi.wikipedia.org/wiki/Subversion)) và bổ sung thêm bộ công cụ mới như biên tập và thiết kế trực quan cho các [miền ngôn ngữ cụ thể](https://vi.wikipedia.org/w/index.php?title=Mi%E1%BB%81n_ng%C3%B4n_ng%E1%BB%AF_c%E1%BB%A5_th%E1%BB%83&action=edit&redlink=1) hoặc bộ công cụ dành cho các khía cạnh khác trong [quy trình phát triển phần mềm](https://vi.wikipedia.org/wiki/Quy_tr%C3%ACnh_ph%C3%A1t_tri%E1%BB%83n_ph%E1%BA%A7n_m%E1%BB%81m).

* **Visual Studio Code**

Là một trình biên tập lập trình code miễn phí dành cho Windows, Linux và macOS, Visual Studio Code được phát triển bởi Microsoft. Nó được xem là một sự kết hợp hoàn hảo giữa IDE và Code Editor. Visual Studio Code hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio Code cũng cho phép người dùng thay đổi theme, phím tắt, và các tùy chọn khác.

Tính năng:

+ Hỗ trợ nhiều ngôn ngữ lập trình: Visual Studio Code hỗ trợ nhiều ngôn ngữ lập trình như C/C++, C#, PHP, HTML, CSS, JavaScript, … Vì vậy, nó dễ dàng phát hiện và đưa ra thông báo nếu chương chương trình có lỗi.

+ Hỗ trợ đa nền tảng: Các trình viết code thông thường chỉ được sử dụng hoặc cho Windows hoặc Linux hoặc Mac Systems. Nhưng Visual Studio Code có thể hoạt động tốt trên cả ba nền tảng trên.

+ Cung cấp kho tiện ích mở rộng: Trong trường hợp lập trình viên muốn sử dụng một ngôn ngữ lập trình không nằm trong số các ngôn ngữ Visual Studio hỗ trợ, họ có thể tải xuống tiện ích mở rộng. Điều này vẫn sẽ không làm giảm hiệu năng của phần mềm, bởi vì phần mở rộng này hoạt động như một chương trình độc lập.

* **NetBeans IDE**

NetBeans IDE là môi trường phát triển tích hợp và cực kỳ cần thiết cho các lập trình viên, công cụ này có thể hoạt động tốt với rất nhiều nền tảng hệ điều hành khác nhau như Linux, Windows, MacOS,... là một mã nguồn mở cung cấp các tính năng cần thiết nhất nhăm tại ra các ứng dụng web, thiết bị di động, desktop. Nó hỗ trợ rất nhiều những ngôn ngữ lập trình như Python, Ruby, JavaScript, Groovy, C / C + +, và PHP.

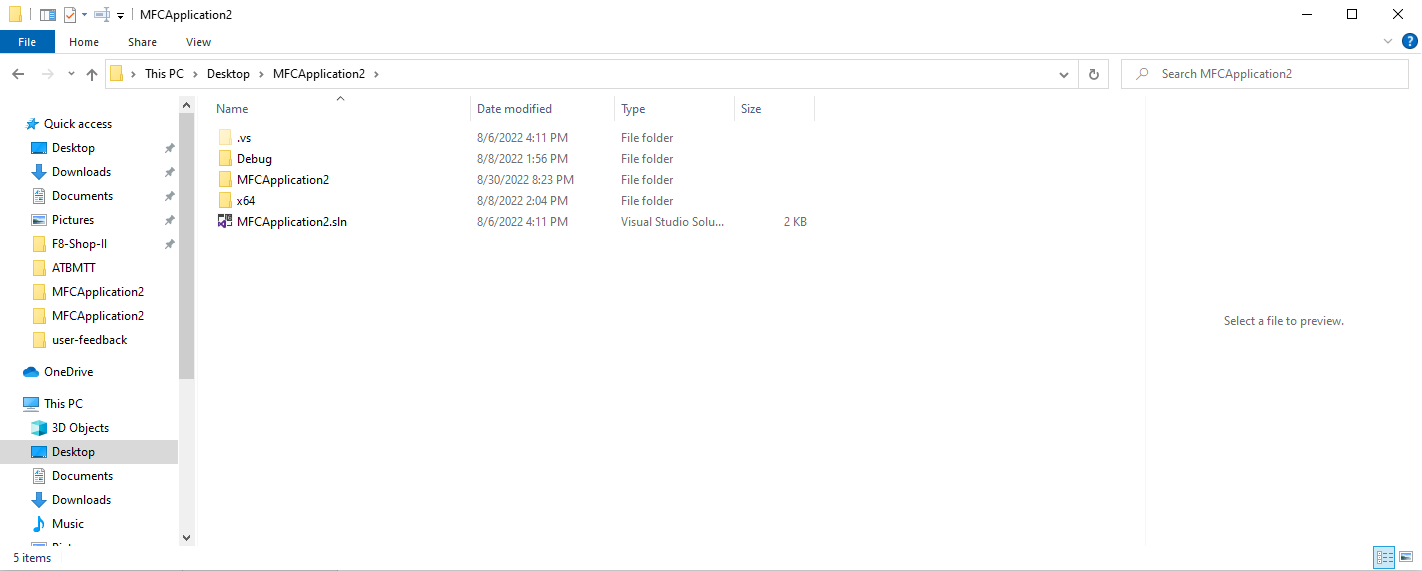
**Tính năng chính của NetBeans IDE:**

* Hỗ trợ nhiều loại ngôn ngữ lập trình.
* Chỉnh sửa mã nguồn thông minh.
* Giao diện trực quan, dễ thao tác, sử dụng,
* Là một công cụ lập trình phần mềm máy tính hoặc phần mềm trên các thiết bị di động.
* Gỡ lỗi mạng nội bộ và từ xa.
* Thử nghiệm tính năng xây dựng giao diện đồ họa.
* Tính năng Quick Search (Tìm kiếm nhanh), tự động biên dịch, hỗ trợ các Framework cho website, trình ứng dụng máy chủ GlassFish và cơ sở dữ liệu.

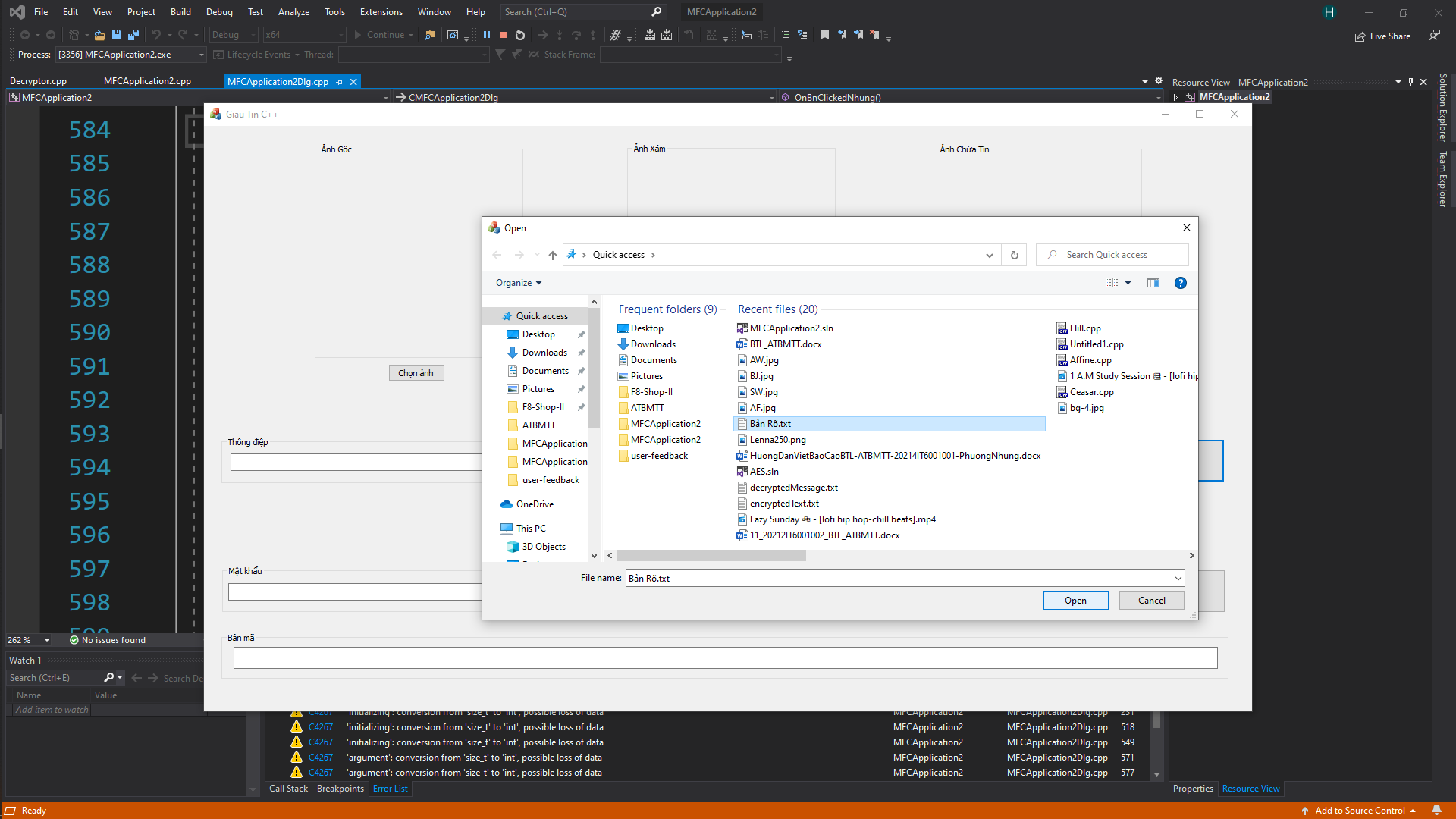
### Cài đặt và chạy chương trình demo

* Cài đặt và chạy chương trình với C++ <Hoàng Anh Nghiệp >

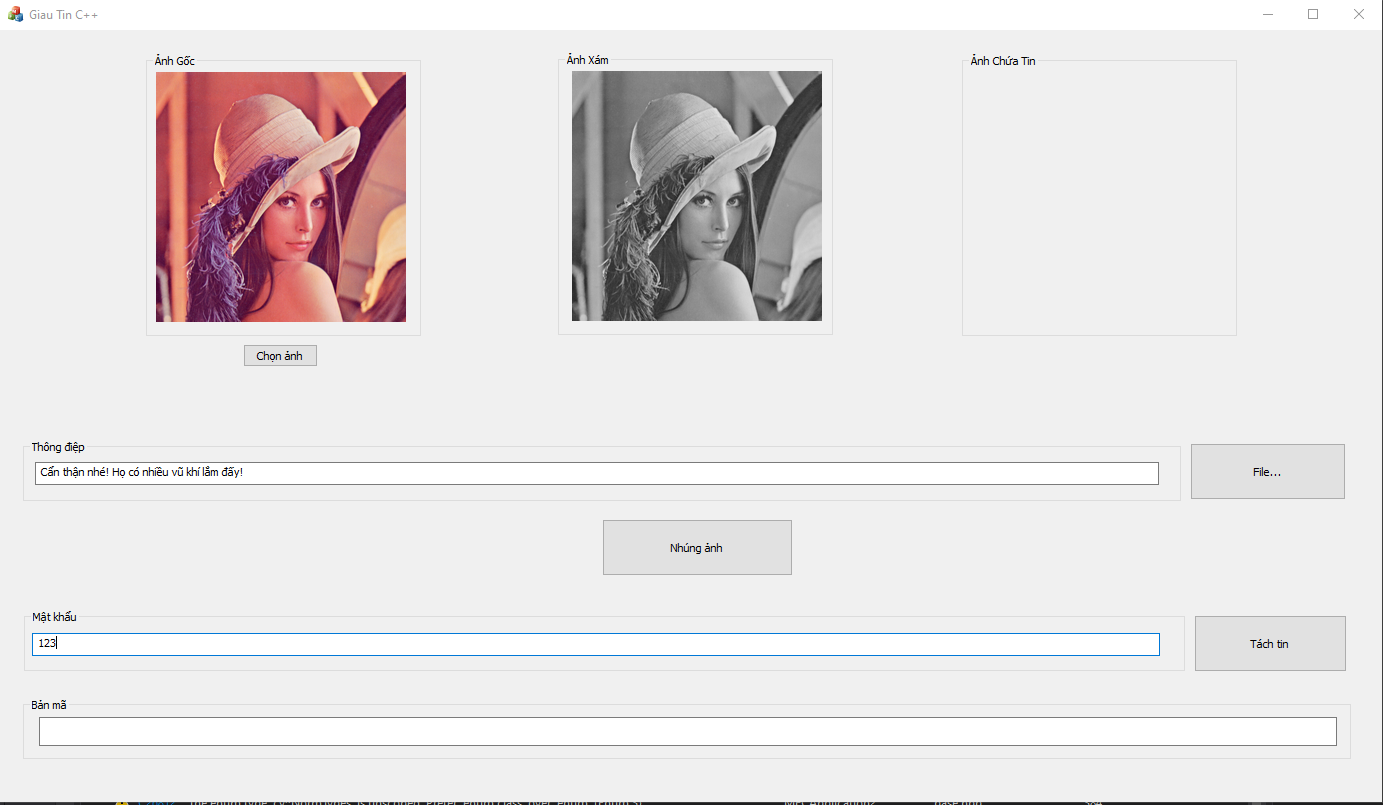
**Bước 1**: Mở file “.sln” của project “MFCApplication2”



**Bước 2:**Chạy chương trình, chọn file để mã hóa

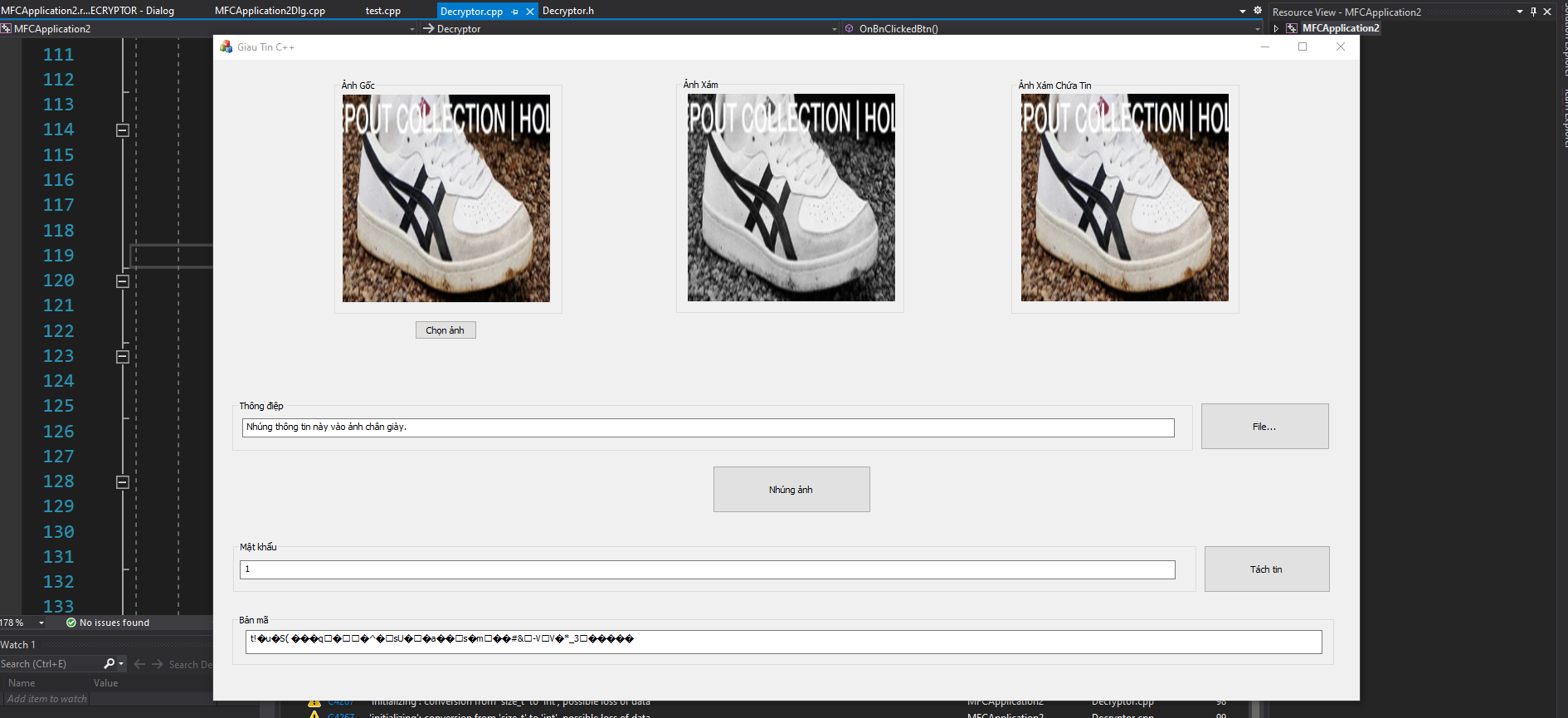


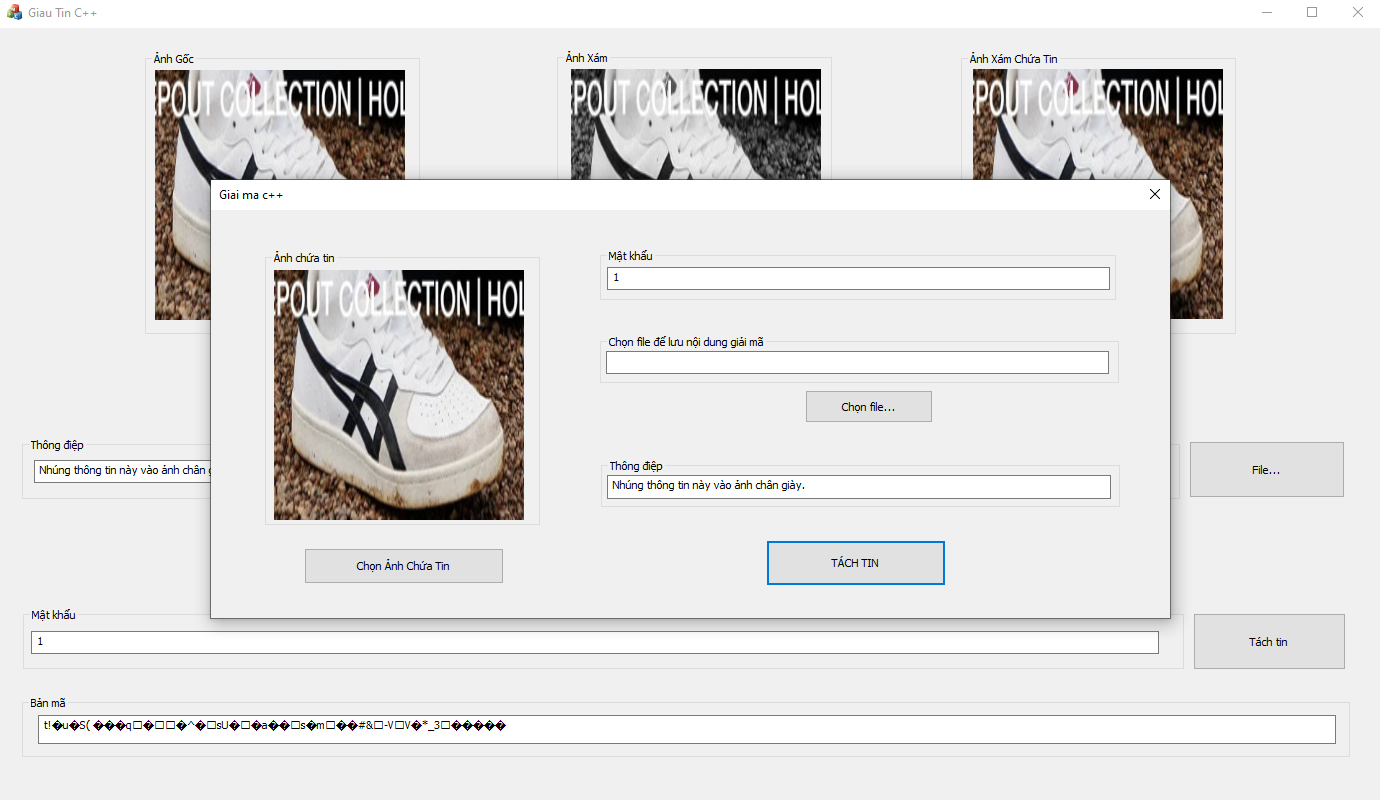
**Bước 3**: Thực hiện theo yêu cầu nhập đầy đủ mật khẩu và chọn ảnh để giấu tin, sau đó nhấn vào “Nhúng ảnh” để thực hiện giấu tin.



Bản mã sẽ hiện ra sau khi “Nhúng ảnh”, nếu muốn tách tin, nhấn “Tách tin”, màn hình để

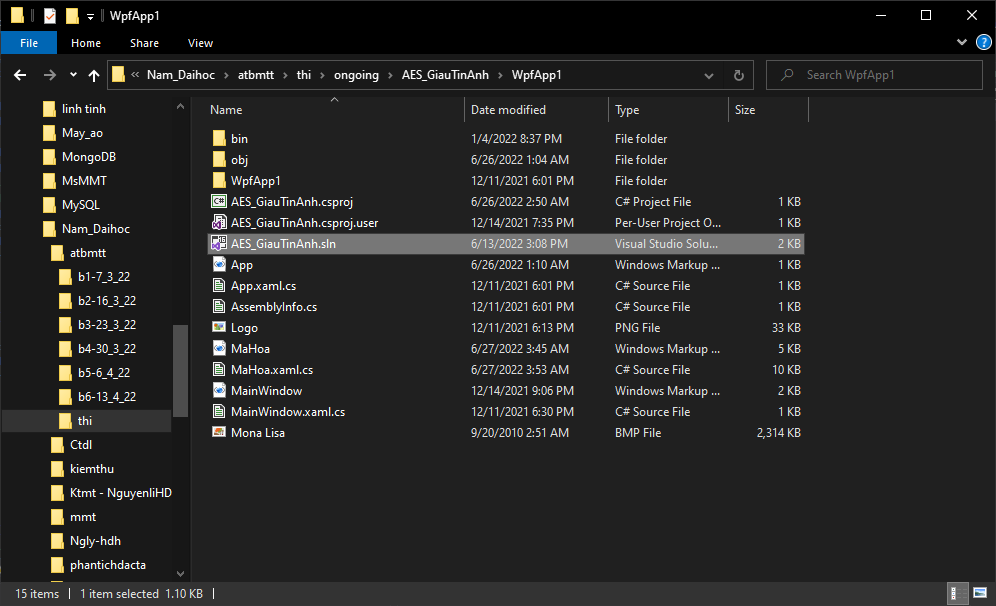
Tách tin sẽ hiện ra:



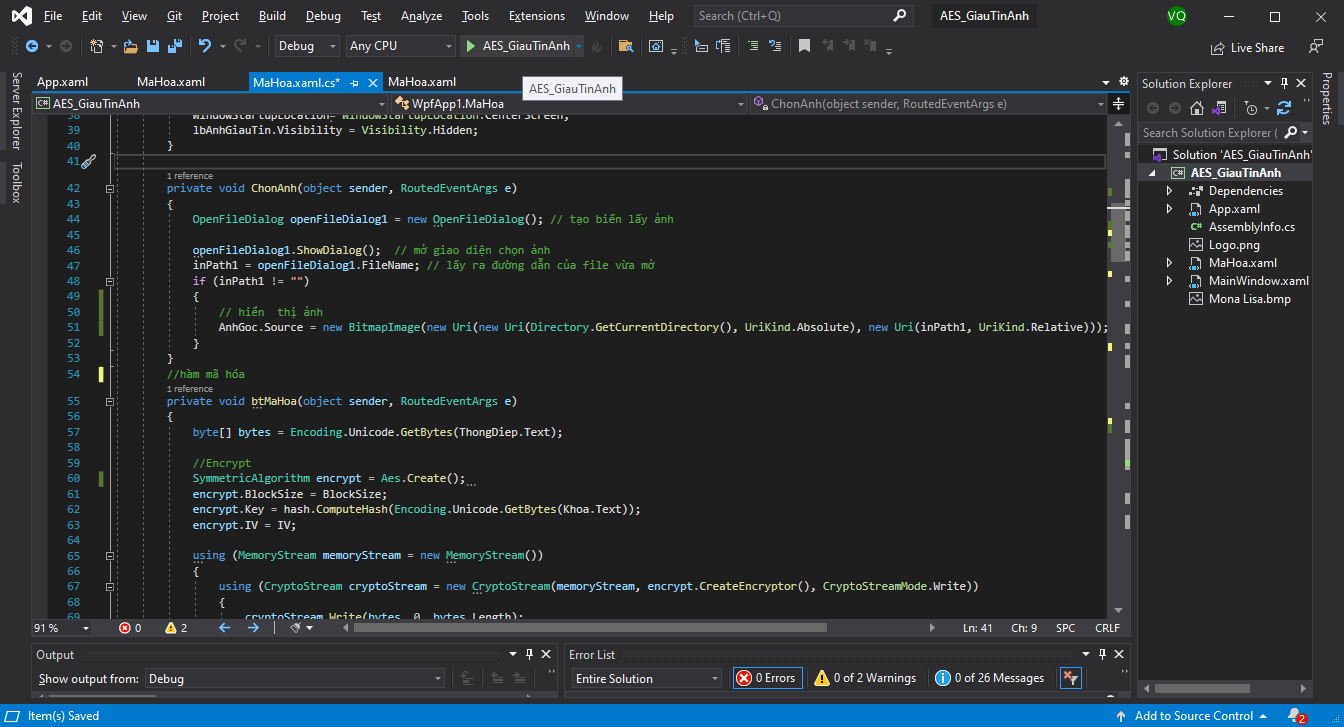


* Cài đặt và chạy chương trình C# <Dương Hải Nguyên>

**Bước 1:** Tại folder project nhấn đúp chuột vào file .sln



**Bước 2:** Cửa sổ Visual Studio khởi động, nhấn tiếp vào nút start project ở thanh công cụ.

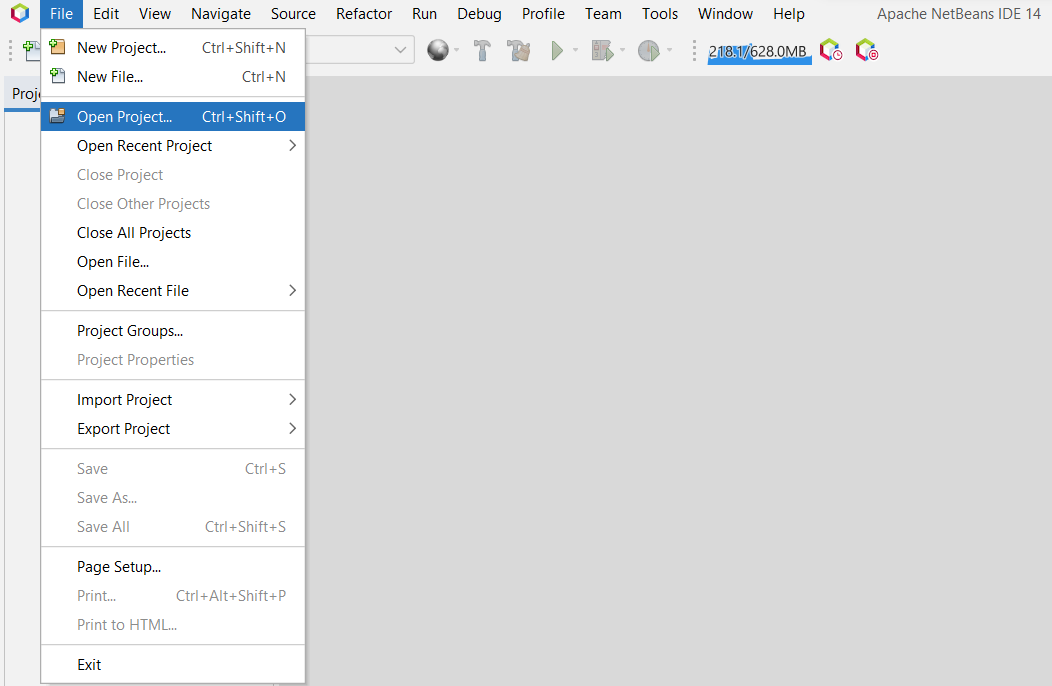


Màn hình chương trình hiện ra như sau:

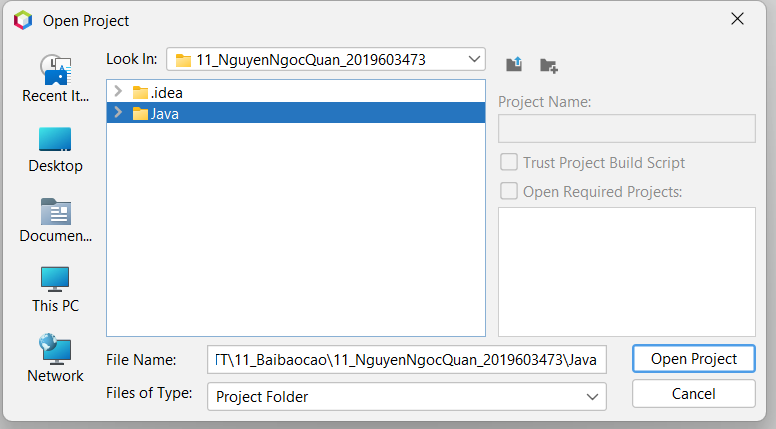


* Cài đặt và chạy chương trình Java <Phạm Văn Nhất>

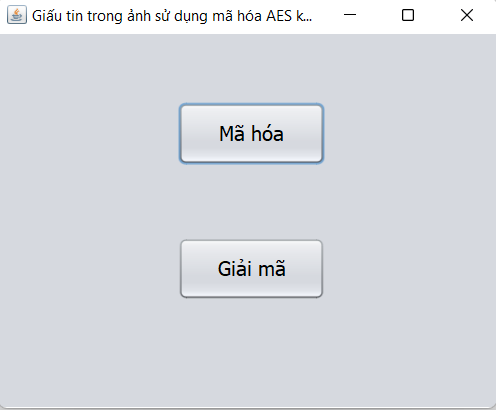
**Bước 1:** Chạy phần mềm NetBeans chọn File -> Chọn open project



**Bước 2:** Chọn đến project Java và chọn Open Project



**Bước 3:** Chọn nút Run ta được màn hình khởi động chương trình

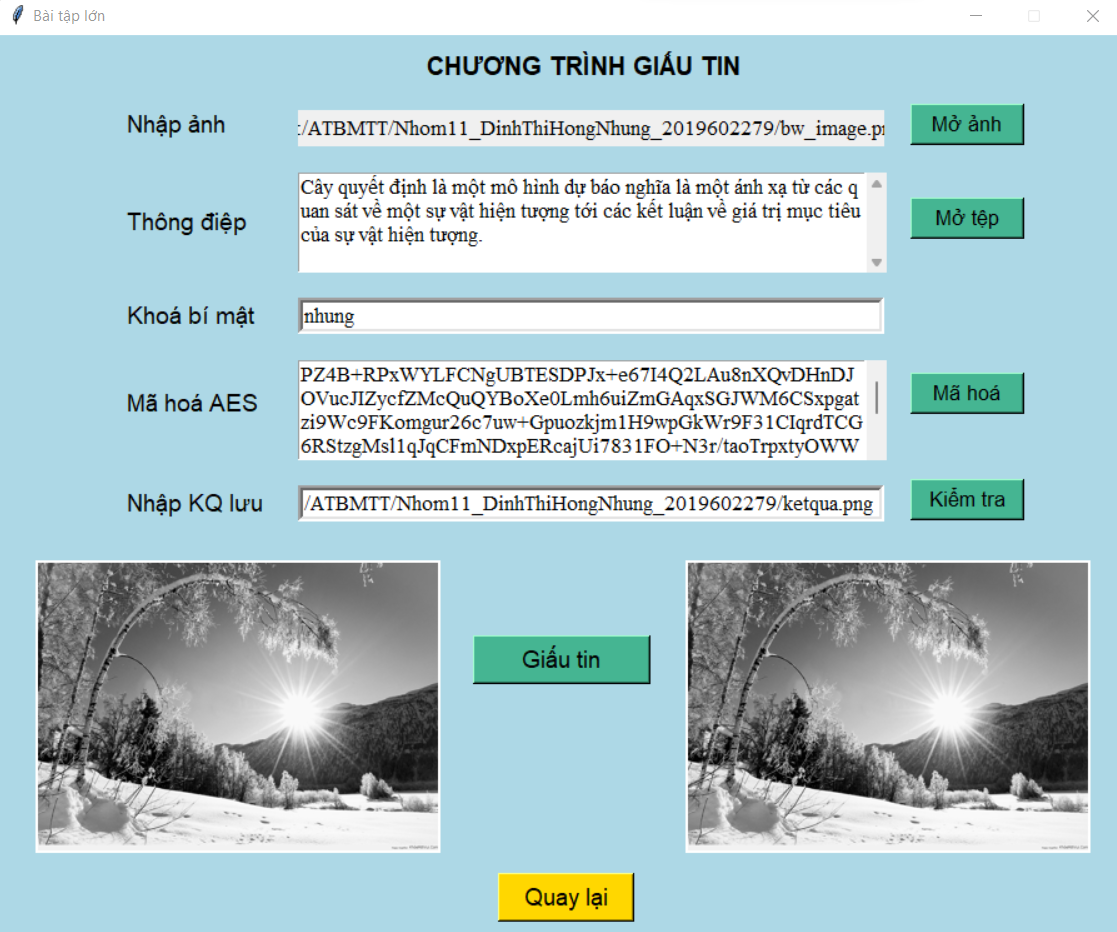


* Cài đặt và chạy chương trình Python <Đinh Thị Hồng Nhung>

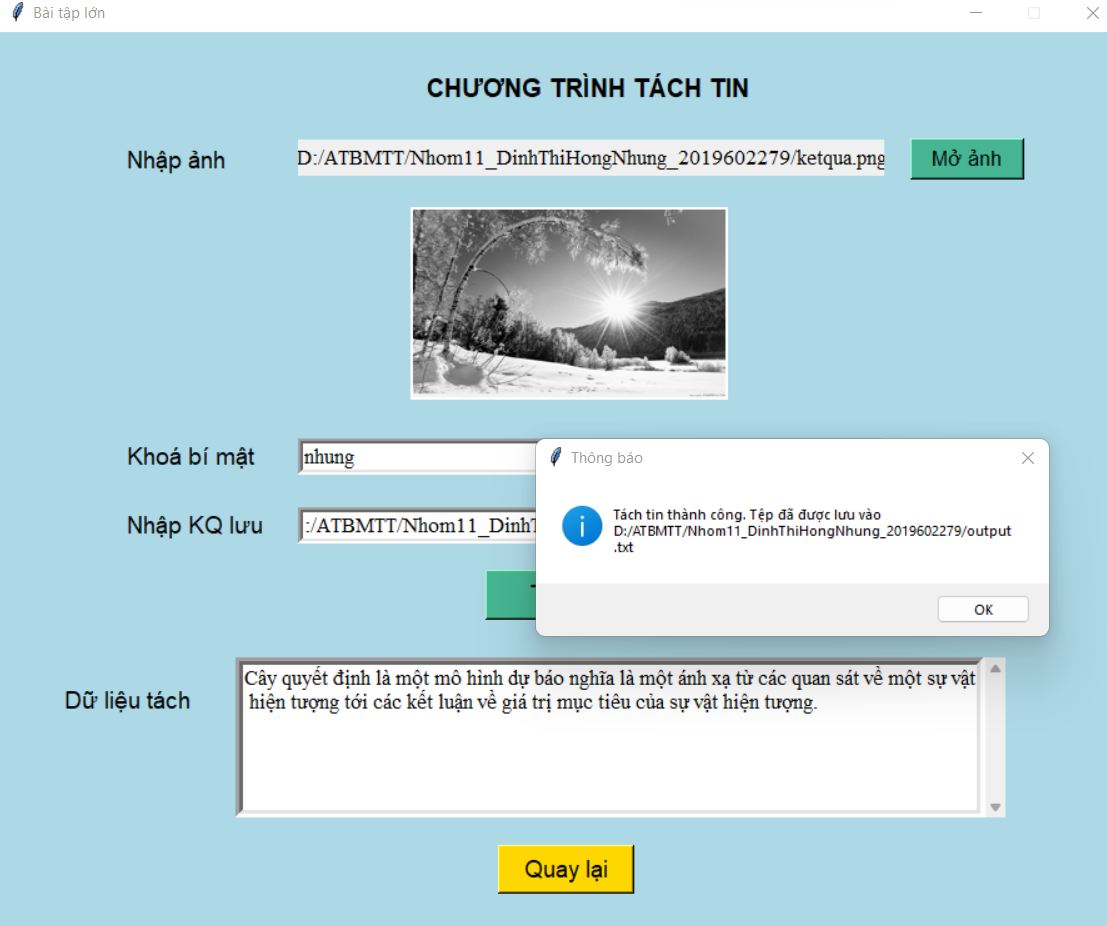
Khởi chạy chương trình



Chọn “Giấu tin” ở menu chính để thực hiện giấu tin



Nếu muốn thực hiện tách tin, chọn “Tách tin” ở menu chính.



## Thực hiện bài toán

### Phân công công việc

|  |  |
| --- | --- |
| Tên sinh viên | Tên công việc |
| Hoàng Anh Nghiệp | - Tìm hiểu về phương pháp đánh giá ảnh trước và sau khi giấu tin  - Tìm hiểu về ngôn ngữ C++ và lập trình giao diện MFC với C++  - Viết chương trình demo với C++ |
| Mạch Văn Quân | - Tìm hiểu về mật mã AES (Giải mã)  - Tìm hiều về ngôn ngữ Ruby  - Viết chương trình demo với Ruby |
| Đinh Thị Hồng Nhung | - Tìm hiểu kỹ thuật giấu tin trong sai phân  - Tìm hiểu về ngôn ngữ Python  - Viết chương trình demo với Python |
| Phạm Văn Nhất | - Tìm hiểu kỹ thuật giấu tin trong ảnh  - Tìm hiểu về ngôn ngữ Java  - Viết chương trình demo với Java |
| Dương Hải Nguyên | - Tìm hiểu cấu trúc ảnh bitmap  - Tìm hiểu về ngôn ngữ C#  - Viết chương trình demo với C# |

### Hoàng Anh Nghiệp – Phương pháp đánh giá ảnh và ngôn ngữ C++

1. **Phương pháp đánh giá ảnh trước và sau khi giấu tin**

PSNR dùng để tính tỉ lệ giữa giá trị năng lượng tối đa của một tín hiệu và năng lượng nhiễu ảnh hưởng đến độ chính xác của thông tin. Bởi vì có rất nhiều tín hiệu có phạm vi biến đổi rộng, nên PSNR thường được biểu diễn đơn vị logarit.

Ngoài ra, PSNR còn được sử dụng để đo chất lượng tín hiệu khôi phục của các thuật toán nén có mất dữ liệu. Tín hiệu trong trường hợp này là dữ liệu gốc và nhiễu là các lỗi xuất hiện khi nén. Khi so sánh các thuật toán nén thường được dự vào sự cảm nhận của con người đối với dữ liệu được khôi phục, chính vì thế trong một số trường hợp dữ liệu được khôi phục của thuật toán này cũng có chất lượng cao hơn các thuật toán khác. PSNR có giá trị càng cao thì chất lượng ảnh càng tốt.

Cách đơn giản nhất là định nghĩa thông qua MSE được dùng cho ảnh 2 chiều có kích thước mxn trong đó I và K là ảnh gốc và được khôi phục tương ứng:



Khi đó, PSNR được tính bởi:



Ở đây, MAXI là giá trị tối đa của điểm ảnh trên ảnh. Khi các điểm ảnh được biểu diễn bởi 8bits, thì giá trị của nó là 255. Trường hợp tổng quát, khi tín hiểu được biểu diễn bởi B bits cho một đơn vị lấy mẫu, thì MAXI là 2B – 1. Trường hợp ảnh máu với 3 giá trị RGB trên một điểm ảnh, cách tính toán cho PSNR tương tự việc tính MSE là tổng của 3 giá trị chia cho kích thước của ảnh và chia cho 3.

Các giá trị thông thường của PSNR trong giấu ảnh và nén video nằm từ 30 đến 50 dB, giá trị càng cao thì càng tốt. Giá trị có thể chấp nhận được khi truyền tín hiệu khâu dây có tổn thất khoảng từ 20dB đến 25dB.

1. **Tìm hiểu C++ và lập trình giao diện với MFC**

**Tổng quan về C++**

C++ là một loại [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) bậc trung (middle-level). Đây là [ngôn ngữ lập trình đa năng](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_%C4%91a_n%C4%83ng) được tạo ra bởi [Bjarne Stroustrup](https://vi.wikipedia.org/wiki/Bjarne_Stroustrup) như một phần mở rộng của [ngôn ngữ lập trình C](https://vi.wikipedia.org/wiki/C_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), hoặc "C với các lớp Class", Ngôn ngữ đã được mở rộng đáng kể theo thời gian và C ++ hiện đại có các tính năng: [lập trình tổng quát](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_t%E1%BB%95ng_qu%C3%A1t), [lập trình hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng), [lập trình thủ tục](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_th%E1%BB%A7_t%E1%BB%A5c), [ngôn ngữ đa mẫu hình](https://vi.wikipedia.org/w/index.php?title=Ng%C3%B4n_ng%E1%BB%AF_%C4%91a_m%E1%BA%ABu_h%C3%ACnh&action=edit&redlink=1) tự do có [kiểu tĩnh](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh#STATIC), [dữ liệu trừu tượng](https://vi.wikipedia.org/w/index.php?title=D%E1%BB%AF_li%E1%BB%87u_tr%E1%BB%ABu_t%C6%B0%E1%BB%A3ng&action=edit&redlink=1), và [lập trình đa hình](https://vi.wikipedia.org/w/index.php?title=L%E1%BA%ADp_tr%C3%ACnh_%C4%91a_h%C3%ACnh&action=edit&redlink=1), ngoài ra còn có thêm các tính năng, công cụ để thao tác với [bộ nhớ cấp thấp](https://vi.wikipedia.org/w/index.php?title=B%E1%BB%99_nh%E1%BB%9B_c%E1%BA%A5p_th%E1%BA%A5p&action=edit&redlink=1). Từ [thập niên 1990](https://vi.wikipedia.org/wiki/Th%E1%BA%ADp_ni%C3%AAn_1990), C++ đã trở thành một trong những ngôn ngữ thương mại ưa thích và phổ biến của lập trình viên.

**C++ dùng để làm gì ?**

C++ được thiết kế hướng tới [lập trình hệ thống](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%E1%BB%87_th%E1%BB%91ng) [máy tính](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh) và [phần mềm nhúng](https://vi.wikipedia.org/wiki/H%E1%BB%87_th%E1%BB%91ng_nh%C3%BAng) trên các mạch [vi xử lý](https://vi.wikipedia.org/wiki/Vi_x%E1%BB%AD_l%C3%BD), bao gồm cả hệ thống có tài nguyên hạn chế và tài nguyên khổng lồ, với ưu điểm vượt trội về hiệu suất, hiệu quả và tính linh hoạt cao, Bao gồm: [phần mềm ứng dụng](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_%E1%BB%A9ng_d%E1%BB%A5ng) [máy tính cá nhân](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh_c%C3%A1_nh%C3%A2n), [trò chơi điện tử](https://vi.wikipedia.org/wiki/Tr%C3%B2_ch%C6%A1i_%C4%91i%E1%BB%87n_t%E1%BB%AD), các hệ thống [máy chủ](https://vi.wikipedia.org/wiki/M%C3%A1y_ch%E1%BB%A7) (ví dụ: phần mềm [thương mại điện tử](https://vi.wikipedia.org/wiki/Th%C6%B0%C6%A1ng_m%E1%BA%A1i_%C4%91i%E1%BB%87n_t%E1%BB%AD), [cỗ máy tìm kiếm](https://vi.wikipedia.org/wiki/M%C3%A1y_truy_t%C3%ACm_d%E1%BB%AF_li%E1%BB%87u) trên web hoặc máy chủ [SQL](https://vi.wikipedia.org/wiki/SQL)) và các ứng dụng ưu tiên về hiệu suất (ví dụ: tổng đài thông tin liên lạc hoặc thiết bị thăm dò không gian).

**Ưu điểm khi sử dụng C++**

Hiệu suất và tốc độ xử lý cực kỳ nhanh.

**Nhược điểm khi sử dụng C++**

**Tràn vùng nhớ**

Như đã đề cập ở trên C++ cho phép bạn tự quản lý vùng nhớ, giúp cho việc tái sử dụng vùng nhớ một cách hiệu quả nhưng nếu quản lý không tốt việc tràn vùng nhớ sẽ xảy ra.

**OOP trong C++ khá phức tạp và khó hiểu**

Trong C++ chúng ta không có khái niệm Interface. Chúng ta có thêm khái niệm hàm thuần ảo, hàm bạn, lớp bạn, hàm hủy (destructor), đa kế thừa,… Do vậy, OOP trong C++ khá phức tạp và khó hiểu. Trong C++ chúng ta phải khai báo phạm vi truy cập của class, phương thức, biến trong file header(file .h) và implement chúng trong file .cpp. Nếu đã quen với OOP trong Java như thì khi chuyển qua C++ sẽ khá bỡ ngỡ vì có một số khái niệm mới.

**Con trỏ là một cái gì đó rất… khó**

Con trỏ là một biến mang địa chỉ của một vùng nhớ mà biến đó trỏ tới. Lập trình viên ban đầu tiếp cận với C++ thì khi học tới con trỏ rất dễ nản vì với những ngôn ngữ bậc cao khác chúng đã lược bỏ khái niệm này để làm ngôn ngữ thân thiện và dễ hiểu hơn. Nhưng con trỏ sinh ra để giúp cho việc thao tác trực tiếp với vùng nhớ, tăng tốc độ thực thi của chương trình nên đôi khi khó học thì cũng đi đôi với lợi ích đó.

1. **Tổng quan về thư viện lập trình giao diện người dùng MFC với C++.**

MFC là viết tắt của Microsoft Foundation Class, là một thư viện C++ để lập trình giao diện người dùng. MFC chứa các lớp định nghĩa sẵn, chúng biểu diễn cách tiếp cận hướng đối tượng đến lập trình Windows và chúng gói các Windows API, tạo nên sự thuận tiện và dễ dàng hơn trong việc sử dụng để tạo ra một sản phẩm so với việc trực tiếp lập trình với win32.

1. **Viết chương trình thực hiện thuật toán AES giấu tin trong ảnh với C++**

**Logic tương tác từ phía người dùng**

**Mã hóa**

void CMFCApplication2Dlg::OnBnClickedNhung()

{

CString inputMessage;

CString inputPassword;

GetDlgItemText(IDC\_EDIT1, inputMessage);

GetDlgItemText(IDC\_EDIT2, inputPassword);

USES\_CONVERSION;

//string EpassWordTemp((CT2CA)inputPassword);

string message((CT2CA)inputMessage);

string passWord((CT2CA)inputPassword);

std::string imgPath((CT2CA)strPath);

imgPath = imgPath.substr(imgPath.find\_last\_of("/\\") + 1);

int lengthMes = message.length();

int lengthPassword = passWord.length();

char encryptedMes[1024];

strcpy\_s(encryptedMes, message.c\_str());

if (lengthMes != 0 && lengthPassword != 0 && imgPath != "") {

Encrypt(encryptedMes, imgPath);

CImage inforImg;

CString newPath(savedPath.c\_str(), savedPath.length());

inforImg.Load(L"grayInfo" + newPath);

CBitmap bitmap;

bitmap.Attach(inforImg.Detach());

inforPic.SetBitmap(bitmap);

isEmbeded = true;

string EpassWordTemp((CT2CA)inputPassword);

Epassword = EpassWordTemp;

string myText;

string encryptedPath = "encryptedText.txt";

ifstream MyReadFile(encryptedPath);

while (getline(MyReadFile, myText)) {

cout << myText;

}

CString textFromFile(myText.c\_str(), myText.length());

SetDlgItemText(IDC\_EDIT3, textFromFile);

MyReadFile.close();

}

else {

MessageBox(\_T("Bạn phải nhập đầy đủ thông tin và chèn ảnh"), \_T("Thiếu thông tin"), MB\_ICONERROR);

}

}

void CMFCApplication2Dlg::OnBnClickedLuu()

{

if (isEmbeded) {

CString inputPassword;

GetDlgItemText(IDC\_EDIT2, inputPassword);

string EpassWordTemp((CT2CA)inputPassword);

Epassword = EpassWordTemp;

Decryptor DecDialog;

DecDialog.DoModal();

}

else {

MessageBox(\_T("Bạn phải nhúng ảnh trước"), \_T("Lỗi"), MB\_ICONERROR);

}

}

void CMFCApplication2Dlg::OnBnClickedChonAnh()

{

int imgWidth = 250;

int imgHeight = 250;

CSize GetBitmapDemension();

CMFCApplication2Dlg Dlg;

CMFCApplication2Dlg ANH = GetDlgItem(IDD\_DIALOG1);

//TODO: Add your control notification handler code here

CFileDialog dialog(true);

INT\_PTR nResult = dialog.DoModal();

if (nResult == IDOK)

{

strPath = dialog.GetPathName();

}

std::string s((CT2CA)strPath);

std::string base\_filename = s.substr(s.find\_last\_of("/\\") + 1);

if (base\_filename.find(".png") != -1 || base\_filename.find(".bmp") != -1) {

Mat imageOCV = imread(s);

resize(imageOCV, imageOCV, Size(imgWidth, imgHeight), 0, 0);

imwrite(base\_filename, imageOCV);

CImage image;

CImage grayImg;

CString newPath(base\_filename.c\_str(), base\_filename.length());

image.Load(newPath);

grayImg.Load(newPath);

CBitmap bitmap\_;

bitmap\_.Attach(image.Detach());

originalPic.SetBitmap(bitmap\_);

int r = 0;

int g = 0;

int b = 0;

float grayPix = 0.0;

COLORREF color;

for (int i = 0; i < imgWidth; i++) {

for (int j = 0; j < imgHeight; j++) {

color = grayImg.GetPixel(i, j);

r = BYTE GetRValue(color);

g = BYTE GetGValue(color);

b = BYTE GetBValue(color);

grayPix = (r + g + b) / 3;

grayImg.SetPixel(i, j, RGB(grayPix, grayPix, grayPix));

}

}

CBitmap grayBitmap\_;

grayBitmap\_.Attach(grayImg.Detach());

grayPIC.SetBitmap(grayBitmap\_);

}

}

void CMFCApplication2Dlg::OnBnClickedButtonTextFile()

{

CString textPath;

CFileDialog newDialog(true);

INT\_PTR nResult = newDialog.DoModal();

if (nResult == IDOK)

{

textPath = newDialog.GetPathName();

}

string myTextPath((CT2CA)textPath);

if (myTextPath.find(".txt") != -1) {

string myText;

ifstream MyReadFile(myTextPath);

while (getline(MyReadFile, myText)) {

cout << myText;

}

CString textFromFile(myText.c\_str(), myText.length());

SetDlgItemText(IDC\_EDIT1, textFromFile);

MyReadFile.close();

}

**Giải mã**

void Decryptor::OnBnClickedBtn()

{

CMFCApplication2Dlg e;

std::string msgFile = "";

std::string msgFile2 = "";

int i1;

std::ifstream readFile;

char TextE;

//get 2 password

CString DpassWord;

GetDlgItemText(DEC\_PASSWORD, DpassWord);

CString EpassWord = e.checkPassword();

std::string savedPath((CT2CA)e.getStrPath());

std::string dePath((CT2CA)e.getStrPath());

savedPath = "z-grayInfo" + savedPath;

dePath = "z-hidden" + dePath;

CString loadedImgPath(savedPath.c\_str(), savedPath.length());

int dotIndex = savedPath.find(".", 0);

int pngVal = savedPath.find(".png", dotIndex);

int bmpVal = savedPath.find(".bmp", dotIndex);

int passwordCmpVal = DpassWord.Compare(EpassWord);

if (passwordCmpVal == 0) {

if (pngVal != -1 || bmpVal != -1) {

e.toDecrypt(dePath);

CImage savedImg;

savedImg.Load(loadedImgPath);

CBitmap bitmap\_;

bitmap\_.Attach(savedImg.Detach());

dec\_infor\_img.SetBitmap(bitmap\_);

readFile.open("decryptedMessage.txt");

if (!readFile) {

MessageBox(\_T("File giải mã không tồn tại."), \_T("Lỗi"), MB\_ICONERROR);

exit(-1);

}

getline(readFile, msgFile);

while (getline(readFile, msgFile) && readFile >> i1) {

readFile.ignore(std::numeric\_limits<std::streamsize>::max(), '\n');

}

readFile.close();

CString finalMsg(msgFile.c\_str(), msgFile.length());

SetDlgItemText(DEC\_MESS, finalMsg);

}

else {

MessageBox(\_T("Ảnh sai định dạng, chọn ảnh có định dạng .bmp hoặc .png"), \_T("Lỗi"), MB\_ICONERROR);

}

}

else {

MessageBox(\_T("Nhập sai mật khẩu, xin mời nhập lại"), \_T("Lỗi"), MB\_ICONERROR);

}

}

### Mạch Văn Quân – Tìm hiểu về mật mã AES và ngôn ngữ Ruby

1. **Hệ mật mã AES**

* **Quy trình giải mã:**

Quá trình giải mã được thực hiện theo chiều ngược lại với quy trình mã hóa, đồng thời các phép biến đổi trong quá trình này cũng được thực hiện đảo ngược. Ngoại trừ phép biến đổi AddRoundKey() không thay đổi vì chính bản thân nó là một phép biến đổi thuận ngịch do chỉ áp dụng một phép toán XOR.

* InvShiftRows() chính là phép biến đổi ngược của ShiftRows().
* Là phép biến đổi ngược của SubBytes() được thực hiện trên bảng thay thế S-Box là nghịch đảo của S-Box.
* InvMixColumns() là phép biến đổi ngược của MixColumns().

1. **Tổng quan về ngôn ngữ Ruby**

* **Ruby là gì?**

**Ruby** là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) [hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng), có khả năng [phản ứng](https://vi.wikipedia.org/wiki/Reflection_(khoa_h%E1%BB%8Dc_m%C3%A1y_t%C3%ADnh)).

Theotác giả, Ruby chịu ảnh hưởng bởi [Perl](https://vi.wikipedia.org/wiki/Perl), [Smalltalk](https://vi.wikipedia.org/wiki/Smalltalk), [Eiffel](https://vi.wikipedia.org/w/index.php?title=Eiffel_(programming_language)&action=edit&redlink=1), [Ada](https://vi.wikipedia.org/wiki/Ada_(programming_language)) và [Lisp](https://vi.wikipedia.org/w/index.php?title=Lisp_(programming_language)&action=edit&redlink=1). Ruby là [ngôn ngữ thông dịch](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_th%C3%B4ng_d%E1%BB%8Bch) và [hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng). Ruby cung cấp nhiều [mẫu hình lập trình](https://vi.wikipedia.org/wiki/M%E1%BA%ABu_h%C3%ACnh_l%E1%BA%ADp_tr%C3%ACnh), bao gồm [lập trình hàm](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C3%A0m), [hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng), [mệnh lệnh](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_m%E1%BB%87nh_l%E1%BB%87nh), [phản xạ](https://vi.wikipedia.org/w/index.php?title=L%E1%BA%ADp_tr%C3%ACnh_ph%E1%BA%A3n_x%E1%BA%A1&action=edit&redlink=1). Nó sử dụng hệ thống [kiểu biến động](https://vi.wikipedia.org/w/index.php?title=Dynamic_type&action=edit&redlink=1) và tự động [quản lý bộ nhớ](https://vi.wikipedia.org/wiki/Qu%E1%BA%A3n_l%C3%BD_b%E1%BB%99_nh%E1%BB%9B) tự động. **Ruby** là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) [hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng), có khả năng [phản ứng](https://vi.wikipedia.org/wiki/Reflection_(khoa_h%E1%BB%8Dc_m%C3%A1y_t%C3%ADnh)). Theo tác giả, Ruby chịu ảnh hưởng bởi [Perl](https://vi.wikipedia.org/wiki/Perl), [Smalltalk](https://vi.wikipedia.org/wiki/Smalltalk), [Eiffel](https://vi.wikipedia.org/w/index.php?title=Eiffel_(programming_language)&action=edit&redlink=1), [Ada](https://vi.wikipedia.org/wiki/Ada_(programming_language)) và [Lisp](https://vi.wikipedia.org/w/index.php?title=Lisp_(programming_language)&action=edit&redlink=1). Ruby là [ngôn ngữ thông dịch](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_th%C3%B4ng_d%E1%BB%8Bch) và [hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng). Ruby cung cấp nhiều [mẫu hình lập trình](https://vi.wikipedia.org/wiki/M%E1%BA%ABu_h%C3%ACnh_l%E1%BA%ADp_tr%C3%ACnh), bao gồm [lập trình hàm](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C3%A0m), [hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng), [mệnh lệnh](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_m%E1%BB%87nh_l%E1%BB%87nh), [phản xạ](https://vi.wikipedia.org/w/index.php?title=L%E1%BA%ADp_tr%C3%ACnh_ph%E1%BA%A3n_x%E1%BA%A1&action=edit&redlink=1). Nó sử dụng hệ thống [kiểu biến động](https://vi.wikipedia.org/w/index.php?title=Dynamic_type&action=edit&redlink=1) và tự động [quản lý bộ nhớ](https://vi.wikipedia.org/wiki/Qu%E1%BA%A3n_l%C3%BD_b%E1%BB%99_nh%E1%BB%9B) tự động.

* **Ngôn ngữ lập trình Ruby làm được những gì?**

Kể từ khi ra đời đến hiện nay, ngôn ngữ Ruby luôn được cải tiến và phục vụ nhu cầu phát triển các ứng dụng phục vụ cho nền kinh tế đang ngày một đa dạng. Ruby là Server-Side Programming Languages, mang đến sự linh hoạt, tốc độ, hiệu quả**.**

* **Ưu điểm và nhược điểm của Ruby**

**+ Ưu điểm của Ruby**

* Ruby là loại ngôn ngữ lập trình vô cùng dễ học, nếu như bạn thành thạo Ruby thì việc sử dụng Ruby on Rails quá đơn giản.
* Nhu cầu công việc liên quan đến lập trình viên Ruby ngày càng tăng cao.
* Hệ sinh thái của Ruby vô cùng rộng lớn và mang lại nhiều hữu ích.
* Ruby là ngôn ngữ lập trình có thể dễ dàng kết nối cùng với **DB2**, **MySQL**, **Oracle** và **Sybase**.
* Bạn có thể thực hiện viết thư viện bền ngoài bằng Ruby hoặc bằng C.
* Tính năn bảo mật của Ruby vô cùng cao và tốt.
* Cú pháp thực hiện Ruby đang ngày càng trở nên linh hoạt rất nhiều.
* Hệ thống chuỗi xử lý của Ruby đang rất mạnh mẽ.
* Ruby sở hữu trình gỡ lỗi vô cùng lớn.
* Lập trình viên Ruby có thể hoạt động trên nhiều nền tảng khác nhau.
* Bạn có thể ứng dụng Ruby để có thể thực hiện nhúng vào **HTML**.
* Ruby là ngôn ngữ sở hữu mã nguồn mở.
* Ngôn ngữ cấp cao.
* OOP tinh khiết (Lập trình hướng đối tượng).

+ **Nhược điểm của Ruby**

* Cách sử dụng Ruby được các chuyên gia đánh giá là vô cùng đơn giản, tuy nhiên quá trình học Ruby có thể gặp nhiều khó khăn ở phần cài đặt môi trường.
* Hiện nay, thời gian xử lý của Ruby thường chậm hơn rất nhiều so với các loại ngôn ngữ lập trình khác.

1. **Viết chương trình demo AES giấu tin ảnh với ngôn ngữ Ruby**

* **Mã nguồn chức năng chính của chương trình:**

|  |
| --- |
| <?php  trait AESModel{  public function modelEncrypt(){  $plaintext = $\_POST['plaintext'];  $password = $\_POST['password'];  $file = $\_POST['photo'];  $method = 'aes-256-cbc';  $key = substr(hash('sha256', $password, true), 0, 32);  $iv = chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0);  $encrypted = base64\_encode(openssl\_encrypt($plaintext, $method, $key, OPENSSL\_RAW\_DATA, $iv));  $binaryMessage = '' ;  for($i =0 ; $i < mb\_strlen($encrypted) ; ++ $i)  {  $character = ord($encrypted[$i]) ;  $binaryMessage .= str\_pad(decbin($character), 8, '0', STR\_PAD\_LEFT);  }  $binaryMessage .= '00000011';  $img = imagecreatefrompng($file);  $width = imagesx($img);  $height = imagesy($img);  $messagePosition = 0;  for ($y = 0; $y < $height; $y++) {  for ($x = 0; $x < $width; $x++) {  if (!isset($binaryMessage[$messagePosition])) {  break 2;  }  $rgb = imagecolorat($img, $x, $y);  $colors = imagecolorsforindex($img, $rgb);  $red = $colors['red'];  $green = $colors['green'];  $blue = $colors['blue'];  $alpha = $colors['alpha'];  // Convert the blue to binary.  $binaryBlue = str\_pad(decbin($blue), 8, '0', STR\_PAD\_LEFT);  // Replace the final bit of the blue colour with our message.  $binaryBlue[strlen($binaryBlue) - 1] = $binaryMessage[$messagePosition];  $newBlue = bindec($binaryBlue);  // Inject that new colour back into the image.  $newColor = imagecolorallocatealpha($img, $red, $green, $newBlue, $alpha);  imagesetpixel($img, $x, $y, $newColor);    // Advance message position.  $messagePosition++;  }  }    // Save the image to a file.  $Image = $\_POST['newImg'];  $newImage = $Image.".png";  // imagefilter($img, IMG\_FILTER\_GRAYSCALE);  imagepng($img, $newImage);  // Destroy the image handler.  imagedestroy($img);  }  public function modelDecrypt(){  $password = $\_POST['password'];  $file = $\_POST['photo'];  $method = 'aes-256-cbc';  $key = substr(hash('sha256', $password, true), 0, 32);  $iv = chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0) . chr(0x0);    $img = imagecreatefrompng($file);  // Read the message dimensions.  $width = imagesx($img);  $height = imagesy($img);  // Set the message.  $binaryMessage = '';  // Initialise message buffer.  $binaryMessageCharacterParts = [];    for ($y = 0; $y < $height; $y++) {  for ($x = 0; $x < $width; $x++) {  // Extract the colour.  $rgb = imagecolorat($img, $x, $y);  $colors = imagecolorsforindex($img, $rgb);  $blue = $colors['blue'];  // Convert the blue to binary.  $binaryBlue = decbin($blue);  // Extract the least significant bit into out message buffer..  $binaryMessageCharacterParts[] = $binaryBlue[strlen($binaryBlue) - 1];  if (count($binaryMessageCharacterParts) == 8) {  // If we have 8 parts to the message buffer we can update the message string.  $binaryCharacter = implode('', $binaryMessageCharacterParts);  $binaryMessageCharacterParts = [];  if ($binaryCharacter == '00000011') {  // If the 'end of text' character is found then stop looking for the message.  break 2;  }  else {  // Append the character we found into the message.  $binaryMessage .= $binaryCharacter;  }  }  }  }  // Convert the binary message we have found into text.  $encrypted = '';  for ($i = 0; $i < strlen($binaryMessage); $i += 8) {  $character = mb\_substr($binaryMessage, $i, 8);  $encrypted .= chr(bindec($character));  }  $encrypted;  $decrypted = openssl\_decrypt(base64\_decode($encrypted), $method, $key, OPENSSL\_RAW\_DATA, $iv);  return $decrypted;  }  }  ?> |

### Phạm Văn Nhất – Tìm hiểu giấu tin và ngôn ngữ Java

1. **Tổng quan về giấu tin**

* **Định nghĩa:**

Giấu thông tin là kỹ thuật giấu (hoặc nhúng) một lượng thông tin số nào đó vào trong một đối tượng dữ liệu số khác (“giấu tin” nhiều khi không cần phải chỉ hành động giấu cụ thể mà chỉ mang ý nghĩa quy ước).

Các hệ thống giấu tin mật có thể chia thành ba loại như:

+ *Giấu tin thuần tuý (Pure Steganography)*

Một bộ 4 σ(C, M, D, E), trong đó C là tập các phương tiện chứa thông tin cần giấu, M là tập thông điệp cần giấu với |C|≥|M|, E: C×M → C là một hàm nhúng thông điệp M vào phương tiện chứa C và D: C → M là hàm giải tin sao cho D(E(c, m)) = m với mọi m ∈ M, c ∈ C được gọi là một hệ pure Steganography.

+ *Giấu tin dùng khoá bí mật (Secret key Steganography)*

Một bộ năm σ(C, M, K, Dk, Ek), trong đó C là tập các phương tiện chứa thông tin cần giấu, M là tập thông điệp cần giấu với |C|≥|M|, K là một tập khoá bí mật, Ek: C×M×K → C là một hàm nhúng thông điệp M vào phương tiện chứa C sử dụng khoá K và Dk: C × K→ M là hàm giải tin sao cho Dk(Ek(c, m, k), k) = m với mọi m ∈ M, c ∈ C và k ∈ K được gọi là một hệ Secret key Stegangraphy.

+ *Giấu tin dùng khoá bí mật (Secret key Steganography)*

Giống như là hệ mã mật khoá công khai, hệ giấu tin mật khoá công khai không sử dụng việc truyền khoá bí mật mà sử dụng hai khoá là khoá bí mật và khoá công khai. Khoá công khai được lưu trong cơ sở dữ liệu công cộng. Được sử dụng trong quá trình giấu tin. Khoá bí mật được sử dụng trong quá trình giải tin.

* **Mục đích của giấu tin:**

+ Bảo mật cho những dữ liệu được giấu.

+ Bảo đảm an toàn (bảo vệ bản quyền) cho chính các đối tượng chứa dữ liệu giấu trong đó.

* **Giấu tin trong ảnh:**

Giấu thông tin trong ảnh, hiện nay, là một bộ phận chiếm tỉ lệ lớn nhất trong các chương trình ứng dụng, các phần mềm, hệ thống giấu tin trong đa phương tiện bởi lượng thông tin được trao đổi bằng ảnh là rất lớn và hơn nữa giấu thông tin trong ảnh cũng đóng vai trò hết sức quan trọng trong hầu hết các ứng dụng bảo vệ an toàn thông tin như: nhận thực thông tin, xác định xuyên tạc thông tin, bảo vệ bản quyền tác giả, điều khiển truy cập, giấu thông tin mật,…

Thông tin sẽ được giấu cùng với dữ liệu ảnh nhưng chất lượng ảnh ít thay đổi và chẳng ai biết được đằng sau ảnh đó mang những thông tin có ý nghĩa. Ngày nay, khi ảnh số đã được sử dụng rất phổ biến, thì giấu thông tin trong ảnh đã đem lại rất nhiều những ứng dụng quan trọng trên nhiều lĩnh vực trong đời sống xã hội. Và một đặc điểm của giấu thông tin trong ảnh đó là thông tin được giấu trong ảnh một cách vô hình, nó như là một cách mà truyền thông tin mật cho nhau mà người khác không thể biết được bởi sau khi giấu thông tin thì chất lượng ảnh gần như không thay đổi đặc biệt đối với ảnh màu hay ảnh xám.

1. **Tổng quan về ngôn ngữ Java**

* **Java là gì?**

Java là một ngôn ngữ lập lập trình, được phát triển bởi Sun Microsystem vào năm 1995, là ngôn ngữ kế thừa trực tiếp từ C/C++ và là một ngôn ngữ lập trình hướng đối tượng**.**

* **Ứng dụng của Java**
* Phát triển ứng dụng cho các thiết bị điện tử thông minh, các ứng dụng cho doanh nghiệp với quy mô lớn.
* Tạo các trang web có nội dung động *(web applet*), nâng cao chức năng của server.
* Phát triển nhiều loại ứng dụng khác nhau: Cơ sở dữ liệu, mạng, Internet, viễn thông, giải trí,...
* **Những đặc điểm của Java**
* Đơn giản và quen thuộc: Vì Java kế thừa trực tiếp từ C/C++ nên nó có những đặc điểm của ngôn ngữ này, Java đơn giản vì mặc dù dựa trên cơ sở C++ nhưng Sun đã cẩn thận lược bỏ các tính năng khó nhất của của C++ để làm cho ngôn ngữ này dễ sử dụng hơn.
* Hướng đối tượng và quen thuộc.
* Mạnh mẽ và an toàn.
* Kiến trúc trung lập, độc lập nền tảng và có tính khả chuyển (Portability).
* Hiệu suất cao.
* Máy ảo (biên dịch và thông dịch).
* Phân tán.
* Đa nhiệm: Ngôn ngữ Java cho phép xây dựng trình ứng dụng, trong đó nhiều quá trình có thể xảy ra đồng thời. Tính đa nhiệm cho phép các nhà lập trình có thể biên soạn phần mềm đáp ứng tốt hơn, tương tác tốt hơn và thực hiện theo thời gian thực.

1. **Viết chương trình demo AES giấu tin ảnh với ngôn ngữ Java**

* **Mã nguồn chức năng chính của chương trình:**

|  |
| --- |
| private void btnEncryptActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  // String content = txtContent.getText().trim();  String content = contentFile.trim();  String secretKey = txtSecretKey.getText().trim();  if (content.compareTo("") == 0 || secretKey.compareTo("") == 0) {  JOptionPane.showMessageDialog(this, "Nội dung và khóa bí mật không được để trống","Dữ liệu không hợp lệ", JOptionPane.ERROR\_MESSAGE);  return;  }  if (bufferedImageBefore == null) {  JOptionPane.showMessageDialog(this, "Bạn chưa chọn ảnh", "Dữ liệu không hợp lệ", JOptionPane.ERROR\_MESSAGE);  return;  }  String encryptedContent = AES.encrypt(content, secretKey);  int[] bits = LSB.getBitFromContent(encryptedContent);  try {  LSB.hideTheMessage(bits, bufferedImageBefore);  File file = new File(".\\images\\image\_encrypted.png");  BufferedImage bufferedImage = ImageIO.read(file);  JLabel l = new JLabel(new ImageIcon(bufferedImage));  imageAffter.getViewport().add(l);  JOptionPane.showMessageDialog(this, "Mã hóa thành công.\n Ảnh được lưu vào :\n" + dirFile.getAbsolutePath()+ "\\image\_encrypted.png", "Mã hóa", JOptionPane.INFORMATION\_MESSAGE);  } catch (Exception e) {  JOptionPane.showMessageDialog(this, "Mã hóa thất bại", "Mã hóa", JOptionPane.ERROR\_MESSAGE);  }  }  private void btnOpenContentActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  try {  String dataString = "";  File file = showFileContentDialog(true);  //txtContent.setText(file.getAbsolutePath());  Scanner scanner = new Scanner(Paths.get(file.getAbsolutePath()), "UTF-8");  while (scanner.hasNext()) {  String data = scanner.nextLine();  dataString += data;  dataString += "\n";  }  // System.out.println(""+dataString);  contentFile = dataString;  txtContent.setText(dataString);  } catch (IOException ex) {  Logger.getLogger(MaHoaForm.class.getName()).log(Level.SEVERE, null, ex);  }  }  private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  String content = contentFile.trim();  String secretKey = txtSecretKey.getText().trim();  String encryptedContent = AES.encrypt(content, secretKey);  jTextField1.setText(encryptedContent);  }  private void btnDecryptActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  String secretKey = txtSecretKey.getText().trim();  if (secretKey.compareTo("") == 0) {  JOptionPane.showMessageDialog(this, "Khóa bí mật không được để trống", "Dữ liệu không hợp lệ", JOptionPane.ERROR\_MESSAGE);  return;  }  if (bufferedImageEncrypted == null) {  JOptionPane.showMessageDialog(this, "Bạn chưa chọn ảnh", "Dữ liệu không hợp lệ", JOptionPane.ERROR\_MESSAGE);  return;  }  try {  String encryptedContent = LSB.decodeTheMessage(bufferedImageEncrypted);  String rawData = AES.decrypt(encryptedContent, secretKey);  if (rawData == null) {  JOptionPane.showMessageDialog(this, "Khóa bí mật không đúng", "Giải Mã", JOptionPane.ERROR\_MESSAGE);  return;  }  txaContent.setText(rawData);  JOptionPane.showMessageDialog(this, "Giải mã thành công", "Giải Mã", JOptionPane.INFORMATION\_MESSAGE);  } catch (Exception e) {  JOptionPane.showMessageDialog(this, "Giải mã thất bại", "Giải Mã", JOptionPane.ERROR\_MESSAGE);  }  } |

### Dương Hải Nguyên – Tìm hiểu cấu trúc ảnh Bitmap và ngôn ngữ C#

1. **Cấu trúc file ảnh Bitmap**

Mỗi file ảnh Bitmap gồm 3 phần như sau:

|  |
| --- |
| **Bitmap Header (54 byte)** |
| **Color Palette** |
| **Bitmap Data** |

* **Bitmap header**

Thành phần bitcount của cấu trúc Bitmap header cho biết số bit dành cho mỗi điểm ảnh và số lượng màu lớn nhất của ảnh.

|  |  |  |
| --- | --- | --- |
| **Byte thứ** | **Ý nghĩa** | **Giá trị** |
| 1-2 | Nhận dạng file | „BM‟ hay 19778 |
| 3-6 | Kích thước file | Kiểu long trong Turbo C |
| 7-10 | Dự trữ | Thường mang giá trị 0 |
| 11-14 | Byte bắt đầu vùng dữ liệu | Offset của byte bắt đầu vùng dữ liệu |
| 15-18 | Số byte cho vùng thông tin | 4 byte |
| 19-22 | Chiều rộng ảnh BMP | Tính bằng pixel |
| 23-26 | Chiều cao ảnh BMP | Tính bằng pixel |
| 27-28 | Số Planes màu | Cố định là 1 |
| 29-30 | Số bit cho 1 pixel (bitcount) | Có thể là 1, 4, 8, 16, 24 tùy theo loại ảnh |
| 31-34 | Kiểu nén dữ liệu | 0: Không nén  1: Nén runlength 8bits/pixel 2: Nén runlength 4bits/pixel |
| 35-38 | Kích thước ảnh | Tính bằng byte |
| 39-42 | Độ phân giải ngang | Tính bằng pixel/metter |
| 43-46 | Độ phân giải dọc | Tính bằng pixel/metter |
| 47-50 | Số màu sử dụng trong ảnh |  |
| 51-54 | Số màu được sử dụng khi  hiện thị ảnh |  |

* **Color pallete**

Bảng màu của ảnh, chỉ những ảnh nhỏ hơn hoặc bằng 8 bit mới có bảng màu.

|  |  |  |
| --- | --- | --- |
| **Địa chỉ (Offset)** | **Tên** | **Ý nghĩa** |
| 0 | RgbBlue | Giá trị cho màu xanh Blue |
| 1 | RgbGreen | Giá trị cho màu xanh Green |
| 2 | RgbRed | Giá trị cho màu đỏ |
| 3 | RgbReserved | Dự trữ |

* **Bitmap data**

Phần này nằm ngay sau phần Palete màu của ảnh BMP. Đây là phần chứa giá trị màu của điểm ảnh trong ảnh BMP. Các dòng ảnh được lưu từ dưới lên trên, các điểm ảnh được lưu từ trái sang phải. Giá trị của mỗi điểm ảnh là một chỉ số trỏ tới phần tử màu tương ứng trong Palete màu.

Đặc điểm nổi bật nhất của định dạng BMP là tập tin hình ảnh thường không được nén bằng bất kỳ thuật toán nào. Khi lưu ảnh, các điểm ảnh được ghi trực tiếp vào tập tin – một điểm ảnh sẽ được mô tả bởi một hay nhiều byte tùy thuộc vào giá trị n của ảnh. Do đó, một hình ảnh lưu dưới dạng BMP thường có kích cỡ rất lớn, gấp nhiều lần so với các ảnh được nén (chẳng hạn GIF, JPEG hay PNG). Định dạng BMP được hỗ trợ bởi hầu hết các phần mềm đồ họa chạy trên Windows, và cả một số ứng dụng chạy trên MS-DOS. Ngay từ Windows 3.1, Microsoft đã cho ra đời phần mềm PaintBrush, một phần mềm hỗ trợ vẽ hình ảnh đơn giản và lưu hình ảnh được vẽ dưới dạng BMP 16 hay 256 màu. Tuy nhiên, do kích thước tập tin ảnh BMP quá lớn, định dạng BMP không phù hợp để trao đổi hình ảnh qua mạng Internet (do hạn chế về tốc độ truyền dữ liệu). Do đó, các trang web thường sử dụng ảnh dạng GIF, JPEG hay PNG. Các định dạng này hỗ trợ các thuật toán nén hình ảnh, vì vậy có thể giảm bớt kích cỡ của ảnh.

1. **Tổng quan về ngôn ngữ C#**

* **C# là gì?**

C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000. C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.

* **Đặc trưng của C#**

**+ C# là ngôn ngữ đơn giản**

C# loại bỏ một vài sự phức tạp và rối rắm của những ngôn ngữ như Java và c++, bao gồm việc loại bỏ những macro, những template, đa kế thừa, và lớp cơ sở ảo (virtual base class).

Ngôn ngữ C# đơn giản vì nó dựa trên nền tảng C và C++. Nếu chúng ta thân thiện với C và C++ hoặc thậm chí là Java, chúng ta sẽ thấy C# khá giống về diện mạo, cú pháp, biểu thức, toán tử và những chức năng khác được lấy trực tiếp từ ngôn ngữ C và C++, nó đã được cải tiến để làm cho ngôn ngữ đơn giản hơn.

**+ C# là ngôn ngữ hiện đại**

Điều gì làm cho một ngôn ngữ hiện đại? Những đặc tính như là xử lý ngoại lệ, thu gom bộ nhớ tự động, những kiểu dữ liệu mở rộng, và bảo mật mã nguồn là những đặc tính được mong đợi trong một ngôn ngữ hiện đại. C# chứa tất cả những đặc tính trên. Nếu là người mới học lập trình có thể chúng ta sẽ cảm thấy những đặc tính trên phức tạp và khó hiểu. Tuy nhiên, cũng đừng lo lắng chúng ta sẽ dần dần được tìm hiểu những đặc tính qua các nội dung khoá học này.

**+ C# là ngôn ngữ lập trình thuần hướng đối tượng**

Lập trình hướng đối tượng (OOP: Object-oriented programming) là một phương pháp lập trình có 4 tính chất. Đó là tính trừu tượng (abstraction), tính đóng gói (encapsulation), tính đa hình (polymorphism) và tính kế thừa (inheritance). C# hỗ trợ cho chúng ta tất cả những đặc tính trên.

C # nó là ngôn ngữ lập trình đa nền tảng. Chúng ta có thể xây dựng các ứng dụng .NET được triển khai trên các nền tảng Windows, Linux và Mac.

1. **Viết chương trình demo AES giấu tin ảnh với ngôn ngữ C#**

* **Mã nguồn chức năng chính của chương trình:**

|  |
| --- |
| //hàm mã hóa  private void btMaHoa(object sender, RoutedEventArgs e)  {  byte[] bytes = Encoding.Unicode.GetBytes(ThongDiep.Text);  //Encrypt  SymmetricAlgorithm encrypt = Aes.Create();  encrypt.BlockSize = BlockSize;  encrypt.Key = hash.ComputeHash(Encoding.Unicode.GetBytes(Khoa.Text));  encrypt.IV = IV;  using (MemoryStream memoryStream = new MemoryStream())  {  using (CryptoStream cryptoStream = new CryptoStream(memoryStream, encrypt.CreateEncryptor(), CryptoStreamMode.Write))  {  cryptoStream.Write(bytes, 0, bytes.Length);  }  ThongDiepMaHoa.Text = Convert.ToBase64String(memoryStream.ToArray());  }  }  //hàm giấu tin  private void GiauTin(object sender, RoutedEventArgs e)  {  Bitmap img = new Bitmap(inPath1);  xamhoa(img);  int i, j;  for (i = 0; i < img.Width; i++)  {  for (j = 0; j < img.Height; j++)  {  System.Drawing.Color pixel = img.GetPixel(i, j);  if (i < 1 && j < ThongDiepMaHoa.Text.Length)  {  Console.WriteLine("R[" + i + "][" + j + "] : " + pixel.R);  Console.WriteLine("R[" + i + "][" + j + "] : " + pixel.B);  Console.WriteLine("R[" + i + "][" + j + "] : " + pixel.G);  char letter = Convert.ToChar(ThongDiepMaHoa.Text.Substring(j, 1));  int value = Convert.ToInt32(letter);  Console.WriteLine("Letter: " + letter + "\n Value: " + value);  img.SetPixel(i, j, System.Drawing.Color.FromArgb(pixel.R, pixel.G, value));  }  if (i == img.Width - 1 && j == img.Height - 1)  {  img.SetPixel(i, j, System.Drawing.Color.FromArgb(pixel.R, pixel.G, ThongDiepMaHoa.Text.Length));  }  }  }  //Luu lai anh da giau tin  SaveFileDialog savefile = new SaveFileDialog();  savefile.Filter = "BMP|\*.bmp|JPG|\*.jpg|PNG|\*.png";  savefile.ShowDialog();  string filename = savefile.FileName;  img.Save(filename);  //Hien thi anh giau tin  inPath1 = filename;  AnhGiauTin.Source = new BitmapImage(new Uri(new Uri(Directory.GetCurrentDirectory(), UriKind.Absolute), new Uri(inPath1, UriKind.Relative)));  lbAnhGiauTin.Visibility = Visibility.Visible;  }  //hàm lấy tin từ ảnh  private void LayThongDiep(object sender, RoutedEventArgs e)  {  if (inPath1 == "")  {  MessageBox.Show("Chưa có ảnh để lấy thông điệp", "Error", MessageBoxButton.OK, MessageBoxImage.Error);  }  else  {  Bitmap img = new Bitmap(inPath1);  string message = "";  System.Drawing.Color lpixel = img.GetPixel(img.Width - 1, img.Height - 1);  int messlen = lpixel.B;  for (int i = 0; i < img.Width; i++)  {  for (int j = 0; j < img.Height; j++)  {  System.Drawing.Color pixel = img.GetPixel(i, j);  if (i < 1 && j < messlen)  {  Console.WriteLine("-------------");  Console.WriteLine("R[" + i + "][" + j + "] : " + pixel.R);  Console.WriteLine("R[" + i + "][" + j + "] : " + pixel.B);  Console.WriteLine("R[" + i + "][" + j + "] : " + pixel.G);  int value = pixel.B;  Console.WriteLine("Value: " + value);  char c = Convert.ToChar(value);  string letter = c.ToString();  message = message + letter;  }  }  }  ThongDiep.Text = message;  }  }  //hàm giải mã  private void btGiaiMa(object sender, RoutedEventArgs e)  {  //Decrypt  byte[] bytes = Convert.FromBase64String(ThongDiep.Text);  SymmetricAlgorithm decrypt = Aes.Create();  decrypt.Key = hash.ComputeHash(Encoding.Unicode.GetBytes(Khoa.Text));  decrypt.IV = IV;  try  {  using (MemoryStream memoryStream = new MemoryStream(bytes))  {  using (CryptoStream cryptoStream = new CryptoStream(memoryStream, decrypt.CreateDecryptor(), CryptoStreamMode.Read))  {  byte[] decryptedBytes = new byte[bytes.Length];  cryptoStream.Read(decryptedBytes, 0, decryptedBytes.Length);  ThongDiepSauGiaiMa.Text = Encoding.Unicode.GetString(decryptedBytes);  }    }  }  catch(Exception ex)  {  MessageBox.Show("Mật khẩu không đúng, hãy nhập lại!", "Error",MessageBoxButton.OK,MessageBoxImage.Error);  }  } |

### Đinh Thị Hồng Nhung – Tìm hiểu giấu tin trên sai phân và ngôn ngữ Python

1. **Giấu tin trên sai phân**

* **Thuật toán giấu tin**
* Đầu vào :
  + - * Ảnh sử dụng để giấu tin.
      * Thông điệp.  Đầu ra :
      * Ảnh đã giấu tin.

**Các bước thực hiện:**

*Bước 1:* Sử dụng kỹ thuật mã hóa AES cho thông điệp cần giấu. Sau đó chuyển chuỗi thông điệp đã mã hóa sang nhị phân.

*Bước 2:* Từ ảnh cấp xám dùng để giấu tin ta có ma trận ảnh tương ứng. Tính giá trị di bằng cách trừ hai điểm ảnh liên tiếp cho nhau di = gi +1 - gi, với gi, gi +1 ∈ [0,. . . , 255], i=1:mxn (với m và n là kích cỡ ảnh).

*Bước 3:* Các giá trị tuyệt đối | di |, | di | ∈ [0,. . . , 255], được phân loại vào các phạm vi Ri (Ri gồm các miền giá trị sau [0 7], [8 15], [16 31], [32 63], [64 127] và [128 255]), với i = 1, 2,. . . n. Cận trên cận dưới và độ rộng của mỗi Ri được ký hiệu lần lượt là li, ui và wi. Giả sử | di | thuộc Rk với k ∈ [1,2. . . 6].

*Bước 4:* Ta xác định được số bit sẽ nhúng theo công thức:

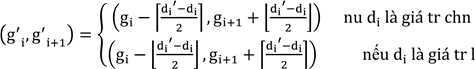


Sau đó chọn n bit từ chuỗi bit cần giấu giấu vào mỗi giá di.

*Bước 5:* Tính lại d được  di’ theo công thức:

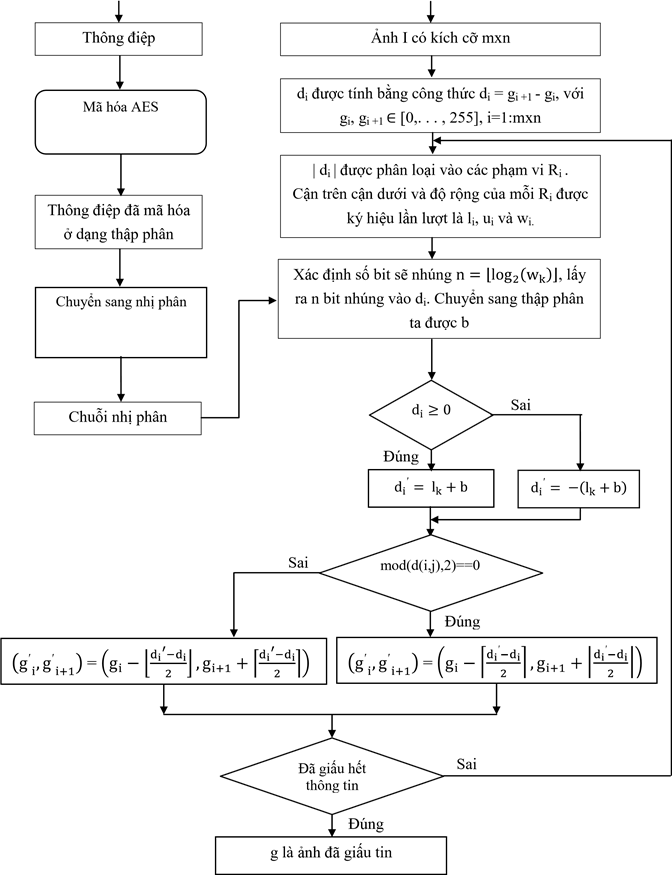


*Bước 6:* Tính các giá trị màu xám mới (gi, gi +1) cho ảnh sau khi giấu tin theo công thức:



Cuối cùng ta được ảnh đã giấu tin.

Sau đây là sơ đồ giấu tin:



* **Thuật toán tách tin**
* Đầu vào :
  + - * Ảnh đã giấu tin.
* Đầu ra :
  + - * Thông điệp đã giấu

**Các bước thực hiện:**

*Bước 1:* Từ ảnh đã giấu tin ta có ma trận ảnh tương ứng. Tính giá trị di bằng cách trừ hai điểm ảnh liên tiếp cho nhau di = gi +1 - gi, với gi, gi +1 ∈ [0,. . . , 255], i=1:mxn (với m và n là kích cỡ ảnh).

*Bước 2:* Xác định phạm vi Ri của mỗi giá trị tuyệt đối | di | được li, ui và wi. Từ đó tính được số bit thông tin đã giấu n =



Giá trị thập phân đã giấu vào di được tính bằng công thức:



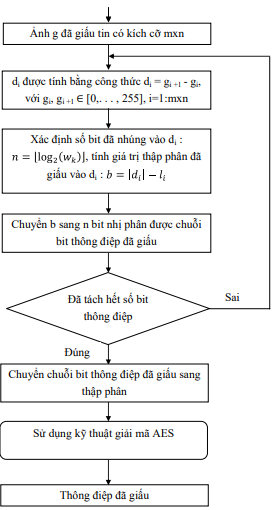
Chuyển b sang n bit nhị phân ta được chuỗi bit thông điệp đã giấu.

*Bước 3:* Lặp lại bước 2 cho đến khi tách được số bit đã giấu.

*Bước 4:* Chuyển chuỗi bit thông điệp đã giấu ở dạng nhị phân sang thập phân ta được chuỗi thông điệp mã hóa đã giấu ở dạng thập phân.

*Bước 5:* Sử dụng kỹ thuật giải mã AES cho chuỗi thông điệp mã hóa đã giấu ta được chuỗi thông điệp cần tách.

Sau đây là sơ đồ thuật toán tách tin:



1. **Tổng quan về ngôn ngữ Python**

Python là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) bậc cao cho các mục đích lập trình đa năng, do [Guido van Rossum](https://vi.wikipedia.org/w/index.php?title=Guido_van_Rossum&action=edit&redlink=1) tạo ra và lần đầu ra mắt vào năm [1991](https://vi.wikipedia.org/wiki/1990). Python được thiết kế với ưu điểm mạnh là dễ đọc, dễ học và dễ nhớ. Python là ngôn ngữ có hình thức rất sáng sủa, cấu trúc rõ ràng, thuận tiện cho người mới học lập trình và là ngôn ngữ lập trình dễ học; được dùng rộng rãi trong phát triển [trí tuệ nhân tạo](https://vi.wikipedia.org/wiki/Tr%C3%AD_tu%E1%BB%87_nh%C3%A2n_t%E1%BA%A1o). Cấu trúc của Python còn cho phép người sử dụng viết mã lệnh với số lần gõ phím tối thiểu. Vào tháng 7 năm 2018, van Rossum đã từ chức lãnh đạo trong cộng đồng ngôn ngữ Python sau 30 năm làm việc.

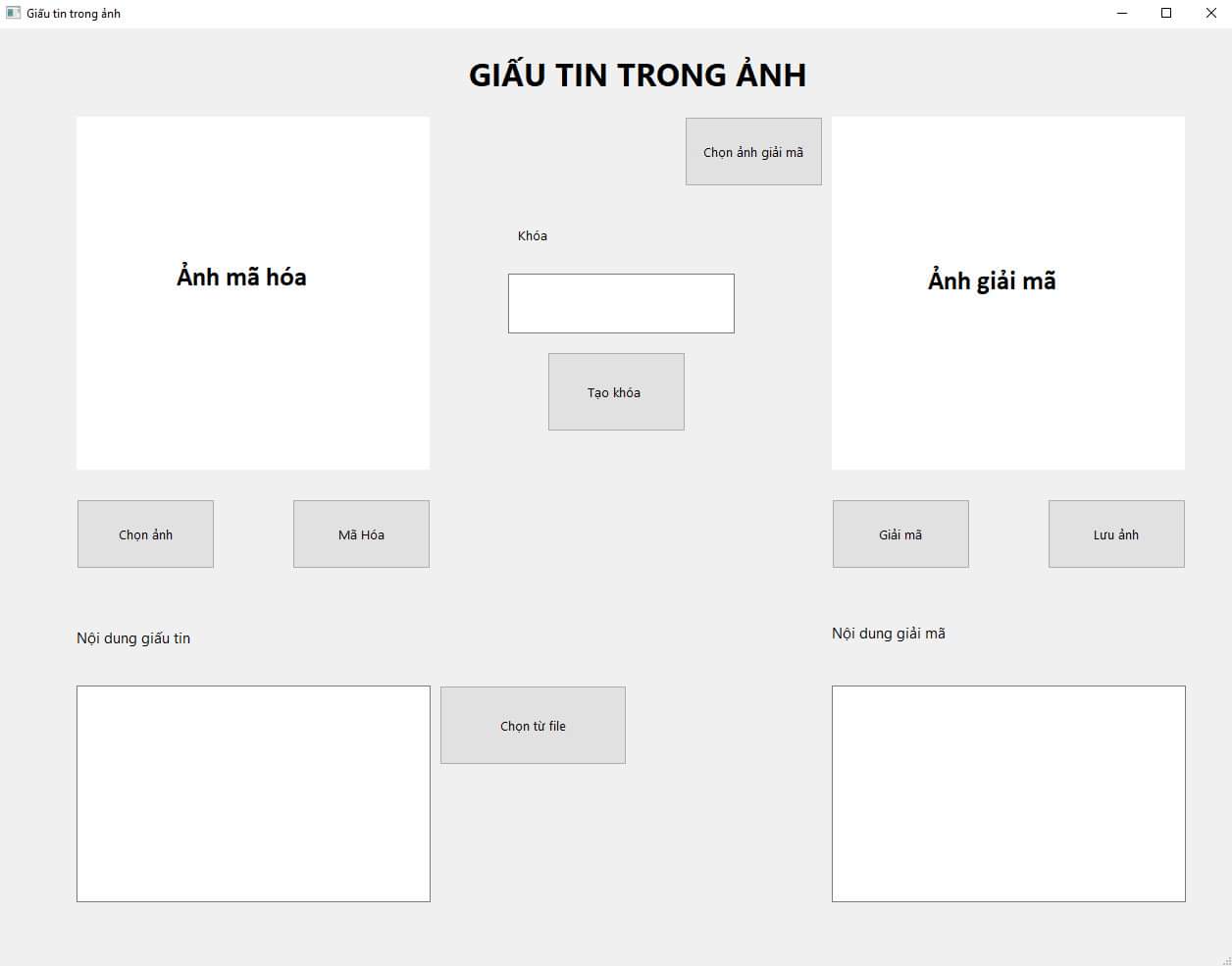
Python hoàn toàn [tạo kiểu động](https://vi.wikipedia.org/w/index.php?title=T%E1%BA%A1o_ki%E1%BB%83u_%C4%91%E1%BB%99ng&action=edit&redlink=1) và dùng cơ chế [cấp phát bộ nhớ tự động](https://vi.wikipedia.org/wiki/Qu%E1%BA%A3n_l%C3%BD_b%E1%BB%99_nh%E1%BB%9B); do vậy nó tương tự như [Perl](https://vi.wikipedia.org/wiki/Perl), [Ruby](https://vi.wikipedia.org/wiki/Ruby_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), [Scheme](https://vi.wikipedia.org/wiki/Scheme), [Smalltalk](https://vi.wikipedia.org/wiki/Smalltalk), và [Tcl](https://vi.wikipedia.org/wiki/Tcl). Python được phát triển trong một dự án mã mở, do tổ chức phi lợi nhuận Python Software Foundation quản lý

Ban đầu, Python được phát triển để chạy trên nền [Unix](https://vi.wikipedia.org/wiki/Unix). Nhưng rồi theo thời gian, Python dần mở rộng sang mọi [hệ điều hành](https://vi.wikipedia.org/wiki/H%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh) từ [MS-DOS](https://vi.wikipedia.org/wiki/MS-DOS) đến [Mac OS](https://vi.wikipedia.org/wiki/Mac_OS), [OS/2](https://vi.wikipedia.org/wiki/OS/2), [Windows](https://vi.wikipedia.org/wiki/Microsoft_Windows), [Linux](https://vi.wikipedia.org/wiki/Linux) và [các hệ điều hành khác thuộc họ Unix](https://vi.wikipedia.org/wiki/T%C6%B0%C6%A1ng_t%E1%BB%B1_Unix). Mặc dù sự phát triển của Python có sự đóng góp của rất nhiều cá nhân, nhưng Guido van Rossum hiện nay vẫn là tác giả chủ yếu của Python. Ông giữ vai trò chủ chốt trong việc quyết định hướng phát triển của Python.

Python luôn được xếp hạng vào những ngôn ngữ lập trình phổ biến nhất.

1. **Viết chương trình demo AES giấu tin ảnh với ngôn ngữ Python**

* **Giao diện chương trình:**



* **Mã nguồn chức năng chính của chương trình:**

|  |
| --- |
| from PyQt5 import QtCore, QtGui, QtWidgets  class Ui\_AppAES(object):      def setupUi(self, AppAES):          AppAES.setObjectName("AppAES")          AppAES.resize(1260, 957)          self.centralwidget = QtWidgets.QWidget(AppAES)          self.centralwidget.setObjectName("centralwidget")          self.imgEncoder = QtWidgets.QLabel(self.centralwidget)          self.imgEncoder.setGeometry(QtCore.QRect(80, 90, 360, 360))          self.imgEncoder.setText("")          self.imgEncoder.setPixmap(QtGui.QPixmap("data\_app/imageEncode-01.png"))          self.imgEncoder.setObjectName("imgEncoder")          self.imgDecode = QtWidgets.QLabel(self.centralwidget)          self.imgDecode.setGeometry(QtCore.QRect(850, 90, 360, 360))          self.imgDecode.setText("")          self.imgDecode.setPixmap(QtGui.QPixmap("data\_app/imageDecode-01.png"))          self.imgDecode.setScaledContents(False)          self.imgDecode.setObjectName("imgDecode")          self.label\_header = QtWidgets.QLabel(self.centralwidget)          self.label\_header.setGeometry(QtCore.QRect(480, 10, 551, 71))          font = QtGui.QFont()          font.setFamily("Segoe UI")          font.setPointSize(24)          font.setBold(True)          font.setWeight(75)          self.label\_header.setFont(font)          self.label\_header.setObjectName("label\_header")          self.btnSelectImg = QtWidgets.QPushButton(self.centralwidget)          self.btnSelectImg.setGeometry(QtCore.QRect(80, 480, 141, 71))          font = QtGui.QFont()          font.setFamily("Segoe UI")          font.setPointSize(10)          font.setBold(False)          font.setWeight(50)          self.btnSelectImg.setFont(font)          self.btnSelectImg.setObjectName("btnSelectImg")          self.btnSaveImg = QtWidgets.QPushButton(self.centralwidget)          self.btnSaveImg.setGeometry(QtCore.QRect(1070, 480, 141, 71))          font = QtGui.QFont()          font.setFamily("Segoe UI")          font.setPointSize(10)          font.setBold(False)          font.setWeight(50)          self.btnSaveImg.setFont(font)          self.btnSaveImg.setObjectName("btnSaveImg")          self.btnEncode = QtWidgets.QPushButton(self.centralwidget)          self.btnEncode.setGeometry(QtCore.QRect(300, 480, 141, 71))          font = QtGui.QFont()          font.setFamily("Segoe UI")          font.setPointSize(10)          font.setBold(False)          font.setWeight(50)          self.btnEncode.setFont(font)          self.btnEncode.setObjectName("btnEncode")          self.btnDecode = QtWidgets.QPushButton(self.centralwidget)          self.btnDecode.setGeometry(QtCore.QRect(850, 480, 141, 71))          font = QtGui.QFont()          font.setFamily("Segoe UI")          font.setPointSize(10)          font.setBold(False)          font.setWeight(50)          self.btnDecode.setFont(font)          self.btnDecode.setObjectName("btnDecode")          self.textEdit = QtWidgets.QTextEdit(self.centralwidget)          self.textEdit.setGeometry(QtCore.QRect(520, 250, 231, 61))          self.textEdit.setObjectName("textEdit")          self.label = QtWidgets.QLabel(self.centralwidget)          self.label.setGeometry(QtCore.QRect(530, 190, 181, 41))          font = QtGui.QFont()          font.setFamily("Segoe UI")          font.setPointSize(10)          self.label.setFont(font)          self.label.setObjectName("label")          self.btnRandom = QtWidgets.QPushButton(self.centralwidget)          self.btnRandom.setGeometry(QtCore.QRect(560, 330, 141, 81))          font = QtGui.QFont()          font.setFamily("Segoe UI")          font.setPointSize(10)          font.setBold(False)          font.setWeight(50)          self.btnRandom.setFont(font)          self.btnRandom.setObjectName("btnRandom")          self.txtEncode = QtWidgets.QTextEdit(self.centralwidget)          self.txtEncode.setGeometry(QtCore.QRect(80, 670, 361, 221))          self.txtEncode.setObjectName("txtEncode")          self.label\_2 = QtWidgets.QLabel(self.centralwidget)          self.label\_2.setGeometry(QtCore.QRect(80, 580, 361, 81))          font = QtGui.QFont()          font.setFamily("Segoe UI")          font.setPointSize(11)          self.label\_2.setFont(font)          self.label\_2.setObjectName("label\_2")          self.btnOpenFile = QtWidgets.QPushButton(self.centralwidget)          self.btnOpenFile.setGeometry(QtCore.QRect(450, 670, 191, 81))          font = QtGui.QFont()          font.setFamily("Segoe UI")          font.setPointSize(10)          font.setBold(False)          font.setWeight(50)          self.btnOpenFile.setFont(font)          self.btnOpenFile.setObjectName("btnOpenFile")          self.label\_3 = QtWidgets.QLabel(self.centralwidget)          self.label\_3.setGeometry(QtCore.QRect(850, 580, 321, 71))          font = QtGui.QFont()          font.setFamily("Segoe UI")          font.setPointSize(11)          self.label\_3.setFont(font)          self.label\_3.setObjectName("label\_3")          self.txtDecode = QtWidgets.QTextEdit(self.centralwidget)          self.txtDecode.setGeometry(QtCore.QRect(850, 670, 361, 221))          self.txtDecode.setObjectName("txtDecode")          AppAES.setCentralWidget(self.centralwidget)          self.statusbar = QtWidgets.QStatusBar(AppAES)          self.statusbar.setObjectName("statusbar")          AppAES.setStatusBar(self.statusbar)          self.retranslateUi(AppAES)          QtCore.QMetaObject.connectSlotsByName(AppAES)      def retranslateUi(self, AppAES):          \_translate = QtCore.QCoreApplication.translate          AppAES.setWindowTitle(\_translate("AppAES", "Giấu tin trong ảnh"))          self.label\_header.setText(\_translate("AppAES", "GIẤU TIN TRONG ẢNH"))          self.btnSelectImg.setText(\_translate("AppAES", "Chọn ảnh"))          self.btnSaveImg.setText(\_translate("AppAES", "Lưu ảnh"))          self.btnEncode.setText(\_translate("AppAES", "Mã Hóa"))          self.btnDecode.setText(\_translate("AppAES", "Giải mã"))          self.label.setText(\_translate("AppAES", "<html><head/><body><p><span style=\" font-size:10pt;\">Khóa</span></p></body></html>"))          self.btnRandom.setText(\_translate("AppAES", "Tạo khóa "))          self.label\_2.setText(\_translate("AppAES", "Nội dung giấu tin"))          self.btnOpenFile.setText(\_translate("AppAES", "Chọn từ file"))          self.label\_3.setText(\_translate("AppAES", "Nội dung giải mã"))  if \_\_name\_\_ == "\_\_main\_\_":      import sys      app = QtWidgets.QApplication(sys.argv)      AppAES = QtWidgets.QMainWindow()      ui = Ui\_AppAES()      ui.setupUi(AppAES)      AppAES.show()      sys.exit(app.exec\_()) |

# Chương 3: Kiến thức lĩnh hội và bài học kinh nghiệm

## Nội dung đã thực hiện

* **Kiến thức và kỹ năng đạt được qua thực hiện đề tài:**

**+** Cách đánh giá ảnh trước và sau khi giấu tin

* Định nghĩa thuật toán giấu tin
* Nghiên cứu tổng quan kỹ thuật giấu tin trong ảnh
* Nghiên cứu cấu trúc ảnh Bitmap
* Hiểu được kỹ thuật giấu tin trong sai phân
* Nắm rõ được mục đích của việc giấu tin trong ảnh
* Nắm chắc được nội dung định nghĩa của thuật toán AES quy trình mã hóa và các bước của quy trình giải mã
* Ứng dụng vào xây dựng chương trình giấu tin trong ảnh sử dụng kết hợp mã hóa AES và kỹ thuật giấu tin trên sai phân
* Nắm vững ngôn ngữ lập trình Matlab 7.7 Java, C#, C++và giải được bài toán có tính ứng dụng vào thực tiễn
* Các công cụ để thực hiện demo chương trình như:Visual Code , Netbeans,…
* **Bài học kinh nghiệm có được sau khi thực hiện đề tài:**

Sau khi hoàn thành bài tập lớn chúng em thấy kỹ thuật giấu tin trong ảnh là hướng nghiên cứu chính của thuật toán giấu thông tin hiện nay và đã đạt được những kết quả khả quan .Bài tập lớn đã trình bày một số khái niệm liên quan đến việc che giấu thông tin trong ảnh số cũng như trình bày kỹ thuật giấu tin trên biên của ảnh nhị phân.

Giấu tin trong ảnh số là một phần của khái niệm giấu thông tin với việc sử dụng ảnh số làm phương tiện mang tin. Giấu thông tin trong ảnh số có nhiều ứng dụng trong thực tế như trong việc xác định quyền sở hữu, chống xuyên tạc thông tin và gửi thông điệp một cách an toàn. Ngày nay, giấu tin trong ảnh nhận được nhiều sự quan tâm của các cá nhân, tổ chức, trường đại học trong việc nghiên cứu về lĩnh vực này.Giấu tin trong ảnh là thủ thuật giấu thông tin cùng với dữ liệu ảnh nhưng chất lượng ảnh lại ít bị thay đổi. Một đặc điểm của giấu thông tin trong ảnh đó là thông tin được giấu trong ảnh một cách vô hình. Dựa vào đặc điểm này ta có thể ứng dụng vào cách truyền thông tin mật cho nhau mà người khác không thể biết được, vì sau quá trình xử lý giấu thông tin thì chất lượng ảnh gần như không thay đổi, đặc biệt đối với ảnh màu.

Giấu tin trong ảnh là kỹ thuật đảm bảo an toàn và bảo mật thông tin. Tính vô hình của thông tin được giấu khái niệm này dựa trên đặc điểm của hệ thống thị giác của con người.Đối với một người có thị giác bình thường sẽ không phân biệt được ảnh trước khi giấu và ảnh sau khi giấu. Trong khi, giấu tin trong ảnh yêu cầu tính vô hình của thông tin giấu ở mức độ cao thì thủy vân lại chỉ yêu cầu ở một cấp độ nhất định. Chẳng hạn như người ta áp dụng thủy vân cho việc gắn một biểu tượng mờ vào một chương trình truyền hình để bảo vệ bản quyền.Số lượng thông tin được giấu lượng thông tin dấu so với kích thước ảnh môi trường cũng là một vấn đề cần quan tâm trong một thuật toán giấu tin. Rõ ràng là có thể chỉ giấu một bit thông tin vào mỗi ảnh mà không cần lo lắng về độ nhiễu của ảnh nhưng như vậy sẽ rất kém hiệu quả khi mà thông tin cần giấu có kích thước lớn. Các thuật toán đều cố gắng đạt được mục đích làm thế nào để giấu được nhiều thông tin nhất mà không gây ra nhiễu đáng kể.

Thuật toán nhúng tin là yếu tố để đánh giá độ an toàn và bảo mật thông tin. Đối với giấu tin trong ảnh thì đây là yêu cầu rất quan trọng. Yêu cầu này dường như khá đơn giản đối với ảnh màu bởi mỗi điểm ảnh được biểu diễn bởi nhiều bit, nhiều giá trị và khi ta thay đổi một giá trị nhỏ nào đó thì chất lượng ảnh thay đổi không đáng kể, thông tin giấu khó bị phát hiện, nhưng đối với ảnh đen trắng mỗi điểm ảnh chỉ là đen hoặc trắng, và nếu ta biến đổi một bit từ trắng thành đen và ngược lại thì sẽ rất dễ bị phát hiện. Do đó, yêu cầu đối với các thuật toán giấu thông tin trong ảnh màu và giấu thông tin trong ảnh đen trắng là khác nhau. Trong khi đối với ảnh màu thì các thuật toán chú trọng vào việc làm sao giấu được càng nhiều thông tin càng tốt thì các thuật toán áp dụng cho ảnh đen trắng lại tập trung vào việc làm thế nào để thông tin giấu khó bị phát hiện nhất.

Với kỹ thuật giấu tin trên biên của ảnh nhị phân thì tính vô hình của thông tin sau khi giấu được đảm bảo .Về mặt lý thuyết thì sau khi đã có lượng thông tin được giấu vào trong ảnh gốc , nó sẽ để lại dù ít,dù nhiều những dấu vết khác so với ảnh gốc ban đầu . Tuy nhiên sau khi thực hiện kỹ thuật giấu tin, quan sát bằng mắt thường thì khó thể nhận ra và phân  biệt được đâu là ảnh gốc và đâu là ảnh mang tin.Nên dùng phương pháp đánh giá PSNR  để đánh giá chất lượng ảnh trước và sau khi giấu tin kết quả PSNR  đạt được là có thể chấp nhận được ,điều đó cho thấy sự biến dạng của ảnh hầu như không có .Như vậy kỹ thuật giấu tin cho chúng ta một kết quả rất triển vọng.

## Hướng phát triển

Tuy nhiên ,giấu tin là một vấn đề phức tạp cộng thêm kinh nghiệm khả năng của chúng em còn hạn chế nên chúng em còn gặp một số khó khăn trong việc tìm hiểu nghiên cứu các kỹ thuật giấu tin trên biên của ảnh nhị phân.

Để hoàn thiện hơn nữa chương trình giấu tin mật kết hợp với mật mã, cần phải nghiên cứu giấu tin trong các loại ảnh khác như: JPEG, TIF, GIF,...Và nghiên cứu tích hợp khả năng trao đổi khóa trên mạng an toàn, bí mật.

Vì thế chúng em rất mong nhận được sự đóng góp ý kiến quý báu của cô đề báo cáo của nhóm chúng em được hoàn thiện hơn.

Chúng em xin cảm ơn.