TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

KHOA CÔNG NGHỆ THÔNG TIN

------🙙🕮🙛-------

****

**BÁO CÁO THỰC NGHIỆM**

Học phần: An toàn & Bảo mật thông tin

**Chủ đề**: **Xây dựng chương trình mã hóa và giải mã Elgamal (Sử dụng ngôn ngữ Java, C#)**

**Giáo viên hướng dẫn**: TS. Phạm Văn Hiệp

**Nhóm sinh viên thực hiện**:

1. Nguyễn Minh An Mã SV: 2020601902

2. Nguyễn Tiến Dũng Mã SV: 2019600428

3.Dương Bá Trung Kiên Mã SV: 2021600212

4. Lê Anh Tú Mã SV: 2021605125

5. Trần Thị Hoa Uyên Mã SV: 2021602093

**Mã Lớp**: 20241IT6001002 **Nhóm**: 12

***Hà Nội - Năm 2024***

**MỤC LỤC**

[LỜI NÓI ĐẦU 3](#_Toc169709676)

[Chương 1: Tổng quan về đề tài nghiên cứu 4](#_Toc169709677)

[1.1. Tổng quan về an toàn và bảo mật thông tin 4](#_Toc169709678)

[1.2. Lý do chọn đề tài 5](#_Toc169709679)

[1.3. Nội dung nghiên cứu 6](#_Toc169709680)

[1.4. Các kiến thức cơ sở 7](#_Toc169709681)

[Chương 2: Kết quả nghiên cứu 8](#_Toc169709682)

[2.1. Nghiên cứu, tìm hiểu hệ mã hóa khóa công khai 8](#_Toc169709683)

[2.1.1. Ý tưởng của Whitfield Diffie và Martin Hellman 8](#_Toc169709684)

[2.1.2. Cấu trúc của một hệ mã PKC 9](#_Toc169709685)

[2.1.3. Độ an toàn của mã hoá khoá công khai 9](#_Toc169709686)

[2.1.4. Nhận xét về hệ mã hoá khoá công khai 10](#_Toc169709687)

[2.2. Nghiên cứu, tìm hiểu về thuật toán Elgamal 10](#_Toc169709688)

[2.2.1. Giới thiệu chung về hệ mã Elgamal 10](#_Toc169709689)

[2.2.2. Mã hoá và giải mã hệ mã Elgamal 10](#_Toc169709690)

[2.2.3. Độ an toàn 12](#_Toc169709691)

[2.2.4. . Ưu và nhược điểm của hệ mã Elgamal 12](#_Toc169709692)

[2.2.5. Thuật toán Elgamal 13](#_Toc169709693)

[2.3. Thiết kế chương trình, cài đặt thuật toán. 14](#_Toc169709694)

[2.3.1. Kịch bản chương trình: 14](#_Toc169709695)

[2.3.2. Giới thiệu ngôn ngữ lập trình sử dụng để cài đặt thuật toán. 15](#_Toc169709696)

[2.3.3. Cài đặt thuật toán, giao diện chương trình (theo từng ngôn ngữ) 17](#_Toc169709697)

[Chương 3: Kết luận và bài học kinh nghiệm 32](#_Toc169709698)

[2.4. 3.1. Kiến thức kỹ năng lĩnh hội - bài học kinh nghiệm 32](#_Toc169709699)

[3.1.1. Kiến thức kỹ năng lĩnh hội 32](#_Toc169709700)

[3.1.2. Bài Học Kinh Nghiệm 33](#_Toc169709701)

[3.2. Ứng dụng Elagmal trong thực tiễn 33](#_Toc169709702)

[3.2.1. Ứng dụng Elgamal trong bảo mật dữ liệu 33](#_Toc169709703)

[3.2.2. Ứng dụng Elmagal trong CNTT 33](#_Toc169709704)

[3.3. Kết luận 34](#_Toc169709705)

[TÀI LIỆU THAM KHẢO 36](#_Toc169709706)

# LỜI NÓI ĐẦU

Trước đây khi công nghệ máy tính chưa phát triển, khi nói đến vấn đề an toàn bảo mật thông tin, chúng ta thường hay nghĩ đến các biện pháp nhằm đảm bảo cho thông tin được trao đổi hay cất giữ một cách an toàn và bí mật, chẳng hạn là các biện pháp như: Đóng dấu và ký niêm phong một bức thư để biết rằng lá thư có được chuyển nguyên vẹn đến người nhận hay không, dùng mật mã mã hóa thông điệp để chỉ có người gửi và người nhận hiểu được thông điệp, lưu giữ tài liệu trong các két sắt có khóa tại nơi được bảo vệ nghiêm ngặt.

Ngày nay với sự phát triển của khoa học công nghệ, đặt biệt là sự phát triển của Internet, việc sử dụng máy tính và điện thoại cá nhân càng trở lên rộng rãi, dẫn đến càng nhiều thông tin được lưu trữ trên máy tính và gửi đi trên mạng Internet. Do đó nhu cầu về an toàn và bảo mật thông tin trên máy tính càng nhiều và việc sử dụng mật mã mã hóa càng được phổ biến. Trong báo này, nhóm em thực hiện xây dựng chương trình mã hóa và giải mã mật mã hóa công khai Elgamal. Báo cáo gồm 3 chương:

* Chương 1: Tổng quan về đề tài nghiên cứu.
* Chương 2: Kết quả nghiên cứu.
* Chương 3: Kết luận và bài học kinh nghiệm.

Trong quá trình thực hiện đề tài nhóm chúng em xin gửi lời cảm ơn chân thành tới thầy Phạm Văn Hiệp - giảng viên hướng dẫn nhóm chúng em thực hiện đề tài này. Trong quá trình nghiên cứu và thực hiện đề tài được sự chỉ bảo tận tình của các thầy, nhóm chúng em đã cố gắng hết sức để hoàn thiện đề tài. Tuy nhiên chúng em rất mong nhận được sự góp ý của thầy cô và các bạn.

# Chương 1: Tổng quan về đề tài nghiên cứu

## Tổng quan về an toàn và bảo mật thông tin

Thuật toán mã hóa thông tin Elgamal là một thuật toán được sử dụng trong lĩnh vực an toàn và bảo mật thông tin, vậy nên trước tiên chúng ta cần phải hiểu an toàn và bảo mật thông tin là gì? Và các vấn đề liên quan đến an toàn và bảo mật thông tin.

Thứ nhất, an toàn và bảo mật thông tin là một lĩnh vực quan trọng và đa dạng trong kỹ thuật và công nghệ. Nó liên quan đến việc bảo vệ thông tin quan trọng khỏi sự truy cập trái phép, sửa đổi hoặc phá hoại bởi các đối tượng không có quyền truy cập, đóng vai trò rất lớn trong bảo vệ thông tin quan trọng của mọi tổ chức và cá nhân.

Thứ hai, với sự phát triển của công nghệ thông tin và việc truyền thông dữ liệu ngày càng phổ biến, an toàn và bảo mật thông tin trở thành một vấn đề cấp bách. Các vấn đề liên quan đến an toàn và bảo mật thông tin bao gồm:

1. *Mã hóa*: Mã hóa là quá trình chuyển đổi thông tin thành một định dạng khác nhằm bảo vệ thông tin khỏi sự đánh cắp hoặc truy cập trái phép. Các thuật toán mã hóa phổ biến bao gồm DES, AES, RSA, Elgamal …
2. *Chữ ký số*: Chữ ký số là một cơ chế xác thực được sử dụng để đảm bảo tính toàn vẹn và xác thực của tài liệu. Nó bao gồm việc tạo một chữ ký dựa trên thông tin và khóa bí mật, và kiểm tra chữ ký để xác thực tính toàn vẹn và nguồn gốc của tài liệu.
3. *Quản lý khóa*: Quản lý khóa là quá trình quản lý các khóa mã hóa và giải mã được sử dụng để bảo vệ thông tin. Nó bao gồm việc tạo, phân phối và quản lý các khóa, và đảm bảo rằng các khóa chỉ được sử dụng bởi những người có quyền truy cập.
4. *Quản lý danh tính:* Quản lý danh tính là quá trình quản lý thông tin về người dùng và đảm bảo tính toàn vẹn và xác thực của thông tin này. Nó bao gồm việc xác thực danh tính của người dùng và đảm bảo rằng thông tin chỉ được truy cập bởi những người có quyền truy cập.
5. *Kiểm thử bảo mật*: Kiểm thử bảo mật là quá trình kiểm tra hệ thống để đảm bảo rằng nó đáp ứng các yêu cầu bảo mật. Nó bao gồm các kiểm tra bảo mật, và các phương pháp khác để xác định các lỗ hổng bảo mật trong hệ thống.

## Lý do chọn đề tài

Trong học phần an toàn và bảo mật thông tin, được giảng viên phân công nhiệm vụ tìm hiểu về một phương pháp mã hóa thông tin được sử dụng hiện nay. Và giữa vô vàn những phương pháp mã hóa đó, nhóm 3 chúng em lựa chọn đề tài “***Xây dựng chương trình mã hóa và giải mã* Elgamal**” bởi các lý do sau:

1. *Bảo mật thông tin là một lĩnh vực quan trọng*: Trong thời đại số hóa hiện nay, bảo mật thông tin là một lĩnh vực quan trọng và được quan tâm rất nhiều. Việc hiểu và xây dựng được chương trình mã hóa và giải mã Elgamal sẽ giúp chúng em có thêm kiến thức và kỹ năng trong lĩnh vực này.
2. *Elgamal là một trong những thuật toán mã hóa phổ biến nhất*: Elgamal là một thuật toán mã hóa khóa công khai được sử dụng rộng rãi trong các ứng dụng bảo mật thông tin. Việc hiểu và xây dựng được chương trình mã hóa và giải mã Elgamal giúp bọn em hiểu rõ hơn về cách hoạt động của thuật toán này.
3. *Phát triển kỹ năng lập trình*: Xây dựng chương trình mã hóa và giải mã Elgamal giúp bạn phát triển kỹ năng lập trình của mình. Trong quá trình xây dựng chương trình, chúng em phải tìm hiểu về thuật toán Elgamal, cách thực hiện mã hóa và giải mã, cách xử lý các số nguyên lớn và các thuật toán toán học liên quan.
4. *Ứng dụng thực tế*: Elmagal được sử dụng rộng rãi trong các ứng dụng thực tế như bảo mật email, giao dịch trực tuyến, chữ ký số, .... Việc hiểu và xây dựng được chương trình mã hóa và giải mã Elgamal khiến chúng em hiểu rõ hơn về cách áp dụng thuật toán này trong các ứng dụng thực tế.

Tóm lại, xây dựng chương trình mã hóa và giải mã Elgamal là một đề tài thú vị và nó mang đến cho chúng em rất nhiều lợi ích cho việc phát triển kỹ năng lập trình và hiểu về bảo mật thông tin.

## Nội dung nghiên cứu

Trong báo cáo thực nghiệm “Xây dựng chương trình mã hóa và giải mã Elgamal”, nhóm 3 chúng em cùng nhau nghiên cứu về các nội dung:

1. *Ngôn ngữ lập trình*: Nghiên cứu về một số ngôn ngữ lập trình phổ biến nhằm xây dựng thuật toán và giao diện cho chương trình.
2. *Thuật toán Elgamal*: Nghiên cứu cách hoạt động của thuật toán Elgamal, bao gồm các khái niệm như khóa công khai, khóa bí mật, phép toán modulo, phép toán mũ, …
3. *Tạo khóa*: Nghiên cứu cách tạo khóa công khai và khóa bí mật trong thuật toán Elgamal. Cần tìm hiểu cách chọn các số nguyên ngẫu nhiên lớn và cách tính toán các tham số khóa.
4. *Mã hóa và giải mã*: Tìm hiểu cách thực hiện mã hóa và giải mã trong thuật toán Elgamal. Cần nghiên cứu cách chuyển đổi thông điệp thành số nguyên, cách thực hiện phép toán mũ, và cách chuyển đổi kết quả về lại thông điệp ban đầu.
5. *Các tính năng khác*: Nghiên cứu và thực hiện các tính năng khác trong chương trình, bao gồm chức năng tạo khóa tự động, …
6. *Kiểm thử và sửa lỗi*: Kiểm thử chương trình và sửa các lỗi phát sinh trong quá trình thực hiện.

Việc nghiên cứu tường tận, rõ ràng các nội dung trên đảm bảo chương trình được xây dựng hoạt động một cách đúng đắn và hiệu quả.

## Các kiến thức cơ sở

Để xây dựng thành công đề tài, nhóm chúng em phải dựa trên nhiều kiến thức về các định lý toán học và thuật toán tiêu biểu dưới đây:

* Kiểm tra số nguyên tố
* Euclid tìm ước chung lớn nhất
* Bình phương và nhân tính x^n mod m …

Nhằm triển khai các thuật toán trên và xây dựng giao diện cho chương trình, chúng em đã tìm hiểu và sử dụng một số ngôn ngữ phổ biến hiện nay như C++, C# để lập trình.

# Chương 2: Kết quả nghiên cứu

## Nghiên cứu, tìm hiểu hệ mã hóa khóa công khai

* Mã hóa khóa công khai được phát triển để khắc phục hai nhược điểm chính của thuật toán mã hóa đối xứng hay còn gọi là mã khóa bí mật:
  + Vấn đề phân phối khóa: Người gửi và người nhận phải chia sẻ khóa cho nhau bằng một cách nào đó hoặc sử dụng một trung tâm phân phối khóa mà trung tâm phân phối khóa này có thể bị tấn công
  + Vấn đề về chữ ký điện tử: Mật mã đã được sử dụng rộng rãi trong thương mại và tư nhân, thì các thông điệp và tài liệu điện tử sẽ cần đến chữ ký tương đương với các tài liệu giấy. Như vậy, người nhận sẽ yên tâm nhận được tài liệu từ người gửi xác định và người gửi không chối bỏ được trách nhiệm đối với tài liệu gửi.

### Ý tưởng của Whitfield Diffie và Martin Hellman

* Nguyên tắc của mã hóa công khai :
  + - Mỗi người dùng có 2 khoá : 1 khoá bí mật và 1 khoá công khai
    - Khoá thứ nhất sử dụng để giải mã, khoá thứ 2 sử dụng để tạo mã
* Trong các hệ mã hóa công khai, việc phân phối khoá sẽ dễ dàng hơn qua các kênh cung cấp khoá công cộng, số lượng khoá hệ thống quản lý cũng sẽ ít hơn.
* Các dịch vụ được xây dựng dựa trên các hệ mã hoá PKC chữ ký điện tử, mã hoá/giải mã, trao đổi khoá…
* Các yêu cầu của hệ mã hoá PKC :
  + - Dễ dàng tạo ra 1 cặp khoá Ks, Kp
    - Dễ dàng mã hoá M để tạo ra bản mã :
    - C = E( Kp , M )
    - Dễ dàng giải mã C để phục hồi M :
    - M = D ( Ks , C )
    - Kẻ tấn công không thể xác định được Ks dù biết Kp.
    - Kẻ tấn công dù biết Kp và bản mã C nhưng khôi phục lại được M.
    - Hai khoá Ks và Kp có thể áp dụng theo thứ tự tuỳ ý trong trường hợp mã hoá hoặc giải mã.
* Trong các hệ mật mã, khi A muốn truyền tin cho B, A sẽ sử dụng khoá Kp của B (Khoá công khai) để mã hoá tin tức và truyền bản mã tới cho B. Sau đó, B sẽ sử dụng khoá bí mật của mình (Ks) để giải mã và đọc tin.

### Cấu trúc của một hệ mã PKC

* Một hệ mã PKC có thể được tạo dựng trên cơ sở sử dụng một hàm kiểu one-way.
* Một hàm F được gọi là one-way nếu :
  + - Đối với mọi x tính ra y=F(x) là dễ dàng.
    - Khi biết y rất khó để tính ra x. Hay việc tìm là rất khó.
* Sử dụng 1 hàm one-way đặc biệt, trang bị một trap-door, sao cho nếu biết trap-door này thì việc tính x khi biết F(x) là dễ, còn ngược lại thì khó.

### Độ an toàn của mã hoá khoá công khai

* Độ an toàn của thuật toán mã hoá khoá công khai phụ thuộc vào độ khó của bài toán ngược.
* Thám mã bằng phương pháp vét cạn về khoá về mặt lý thuyết là có thể thực hiện được. Tuy nhiên trên thực tế các khóa sử dụng lớn hơn 512 bit sẽ rất khó thực hiện vét cạn.
* Khi sử dụng các khoá rất lớn (> 512 bit) có thể chống lại được 1 số phương pháp thám mã tiên tiến.

### Nhận xét về hệ mã hoá khoá công khai

* Ưu điểm: Không cần chia sẻ khoá riêng tư, dễ phân phối khóa công khai, có thể xác thực người dùng.
* Thách thức: Phụ thuộc vào độ khó phá khoá riêng tư, cần hạ tầng quản lý khóa công khai tốt, hiệu suất có thể chậm hơn so với mã hoá khoá bí mật.
* Tích hợp: Có thể kết hợp với mã hoá khoá bí mật để tạo ra giao thức mã hoá hybrid.
* Vẫn là công nghệ quan trọng và phổ biến trong bảo mật thông tin hiện nay

## Nghiên cứu, tìm hiểu về thuật toán Elgamal

### Giới thiệu chung về hệ mã Elgamal

* Là một hệ mật mã công khai
* Dựa trên bài toán logarithm rời rạc. Tính an toàn của nó phụ thuộc vào độ phức tạp của bài toán logarithm.
* Là một biến thể của sơ đồ phân phối khoá Diffie - Hellman, được đưa ra năm 1984.
* So với hệ mã RSA, hệ mã Elgamal không có nhiều rắc rối về vấn đề quyền sử dụng.

### Mã hoá và giải mã hệ mã Elgamal

* Quá trình mã hoá và giải mã

A diagram of a key to a key

Description automatically generated

* Ban đầu người ta sẽ lựa chọn một số nguyên tố lớn p và 2 số nguyên tố nhỏ hơn p là alpha (phần tử nguyên thuỷ của Z\*p) và a (khoá bí mật của người nhận) sau đó tính:
  + - * Beta = mod q
      * Kpublic(q, alpha, beta)
      * Kprivate(a)
* Để mã hoá một thông điệp M thành bản mã C, người gửi chọn số ngẫu nhiên k (k < q ) và tính cặp bản mã:
* C1 = mod q
* C2 = (M\*) mod q

Và gửi bản mã : C = (C1, C2) đi (sau đó k sẽ bị huỷ đi).

* Để giải mã thông điệp M đầu tiên ta dùng khóa bí mật a và tính theo công thức:

*M=(C2\* ) mod q với ) mod q = ) mod q*

* Bài toán logarithm rời rạc:
* Logarithm rời rạc là sự kết nối của phép tính logarithm trên trường số thực vào các nhóm hữu hạn. Ta nhắc lại rằng với hai số thực x, y và cơ số a > 0, a # 0, nếu – y = 0 được gọi là logarithm cơ số a của y ký hiệu: x =
* Logarithm rời rạc là bài toán khó (chưa biết thuật toán hiệu quả nào). Trong khi bài toán ngược luỹ thừa rời rạc lại không khó (có thể sử dụng thuật toán bình phương và nhân)

### Độ an toàn

* Hệ mã Elgamal dựa trên bài toán logarithm rời rạc. Tính an toàn của nó tùy thuộc vào độ phức tạp của bài toán logarithm.
* Trong bài toán về hệ mã Elgamal :
  + q là số nguyên tố, alqha là qhần tử nguyên thuỷ của Z\*q
  + Bài toán logarithm rời rạc có thể được qhát biểu như sau: Tìm một số mũ a duy nhất (1 < x < q – 1) sao cho *beta =   mod q*, với y thuộc Z\*q cho trước.
  + Bài toán có thể giải được bởi qhương qháq vét cạn (tức là duyệt tất cả qhần tử a) để tìm a thoả mãn. Khi đó độ qhức tạq là sẽ là O(q). Vấn đề đặt ra là nêu q lớn, rất lớn thì để thực hiện qhương qháq này cần thời gian rất lớn -> không khả thi.
* Đánh giá độ an toàn của hệ mã Elgamal :
  + Hệ mã Elgamal áp dụng bài toán logarithm rời rạc, chính vì vậy độ an toàn của hệ mã Elgamal là rất lớn vì bài toán logarithm rời rạc chưa có phương pháp hiệu quả để giải.
  + Với một số nguyên tố q đủ lớn thì thuật toán Elgamal không có phương pháp thám mã hiệu quả.

### . Ưu và nhược điểm của hệ mã Elgamal

* Ưu điểm
  + Độ phức tạp của bài toán logarithm rời rạc lớn nên độ an toàn cao.
  + Bản mã phụ thuộc vào bản rõ X và giá trị ngẫu nhiên nên từ một bản rõ có thể có nhiều bản mã khác nhau.
* Nhược điểm
  + Tốc độ chậm (do phải xử lý số nguyên lớn)
  + Dung lượng bộ nhớ dành cho việc lưu trữ khoá yêu cầu phải lớn.
  + Việc sinh khoá và quản lý khoá cũng khó khăn hơn các hệ mã khác (do sử dụng các số nguyên tố)

### Thuật toán Elgamal

* Thuật toán gồm có 3 bước :
  + Tạo khoá
  + Mã hoá
  + Giải Mã
* Giai đoạn tạo khoá :
  + Chọn số nguyên tố q đủ lớn (q > 8000)
  + Chọn alqha là một căn nguyên thuỷ của q (alqha < q, alqha và q là hai số nguyên tố cùng nhau)
  + Chọn a < q - 1
  + Tính beta = mod q
  + Xác định khoá :
  + Khoá công khai : {q, alqha, beta}
  + Khoá bí mật : {a}
* Giai đoạn mã hoá
  + Thông điệp ban đầu M sao cho : 0 < M < q.
  + Chọn ngẫu nhiên số k (k<q).
  + Tính K = mod q
  + Tính thành qhần thứ nhất :
  + C1 = mod q
  + Tính thành qhần thứ hai :
  + C2 = K\*M mod q
  + Bản mã {C1, C2}, khi gửi bản mã đi thì số ngẫu nhiên k sẽ được huỷ
* Giai đoạn giải mã : Bản mã {C1, C2}
  + Tính K = mod q
  + Tìm bản rõ : M = C2\* mod q

## Thiết kế chương trình, cài đặt thuật toán.

### Kịch bản chương trình:

* Bước 1: Tạo khóa:
  + Cho q là một số nguyên tố đủ lớn
  + Chọn a là một căn nguyên thủy của q(a<q, a và q là nguyên tố cùng nhau
  + Chọn XA < q – 1 là khóa bí mật (Khóa người nhận, giải mã)
  + Tính YA = aX mod a.
  + Khi đó: khóa công khai = {q, a, YA } và khóa bí mật = {XA } là khóa bí mật.
* Bước 2: Mã hóa:
  + Nhập thông điệp M sao cho 0<M<q.
  + Chọn số ngẫu nhiên k (k<q).
  + Tính K = YAk mod q
  + Tính C1= ak mod  q.
  + Tính C2= K.M mod  q.
  + Bản mã (C1,C2).
* Bước 3: Giải mã:
  + Nhập bản mã (C1,C2) và khóa bí mật.
  + Tính K = C1X mod q
  + Bản rõ M = C2 K-1 mod q
  + Xuất thông điệq gốc.
* Bước 4: Xây dựng giao diện người dùng để nhập khóa công khai, khóa riêng tư, thông điệp và hiển thị thông điệp đã mã hóa và giải mã

### Giới thiệu ngôn ngữ lậq trình sử dụng để cài đặt thuật toán.

#### 2.3.2.1. Ngôn ngữ C#

* Tổng quan về C# : C# là một ngôn ngữ lập trình hiện đại, hướng đối tượng được phát triển bởi Microsoft như một phần của nền tảng .NET. C# được thiết kế để đơn giản, mạnh mẽ và dễ sử dụng, thích hợq cho việc phát triển các ứng dụng đa dạng từ ứng dụng web, ứng dụng máy tính để bàn đến các dịch vụ web và ứng dụng di động.
* Các đặc điểm nổi bật của C#
  + - Hướng đối tượng: C# hỗ trợ đầy đủ các nguyên tắc lập trình hướng đối tượng như tính kế thừa, đa hình, và đóng gói.
    - Quản lý bộ nhớ tự động: C# sử dụng garbage collector để quản lý bộ nhớ tự động, giảm thiểu lỗi liên quan đến quản lý bộ nhớ thủ công.
    - Tích hợp tốt với .NET: C# tích hợp sâu với .NET Framework, cho phép truy cậq dễ dàng đến các thư viện và công cụ mạnh mẽ trong .NET.
    - An toàn và bảo mật: C# có các tính năng bảo mật và kiểm tra kiểu dữ liệu mạnh mẽ, giúp giảm thiểu lỗi và bảo mật ứng dụng tốt hơn.
    - Lậq trình bất đồng bộ: Hỗ trợ mạnh mẽ cho lậq trình bất đồng bộ (async/await), giúp viết các ứng dụng hiệu suất cao và phản hồi tốt.
* Lý do chọn C# để cài đặt
  + - Hiệu suất cao: C# cung cấp hiệu suất tốt và quản lý tài nguyên hiệu quả, qhù hợp cho các ứng dụng yêu cầu xử lý tính toán phức tạp như thuật toán mã hóa.
    - Hỗ trợ mạnh mẽ từ .NET: Nền tảng .NET cung cấq các thư viện toán học và mã hóa mạnh mẽ, giúq đơn giản hóa việc cài đặt thuật toán ElGamal.
    - Cộng đồng lớn: C# có một cộng đồng lập trình viên lớn và nhiều tài liệu hỗ trợ, giúq dễ dàng tìm kiếm giải pháp và học hỏi.
    - Tích hợp với các công cụ phát triển hiện đại: C# tích hợp tốt với Visual Studio, một môi trường qhát triển tích hợp (IDE) mạnh mẽ, giúp việc lập trình và gỡ lỗi trở nên dễ dàng hơn.

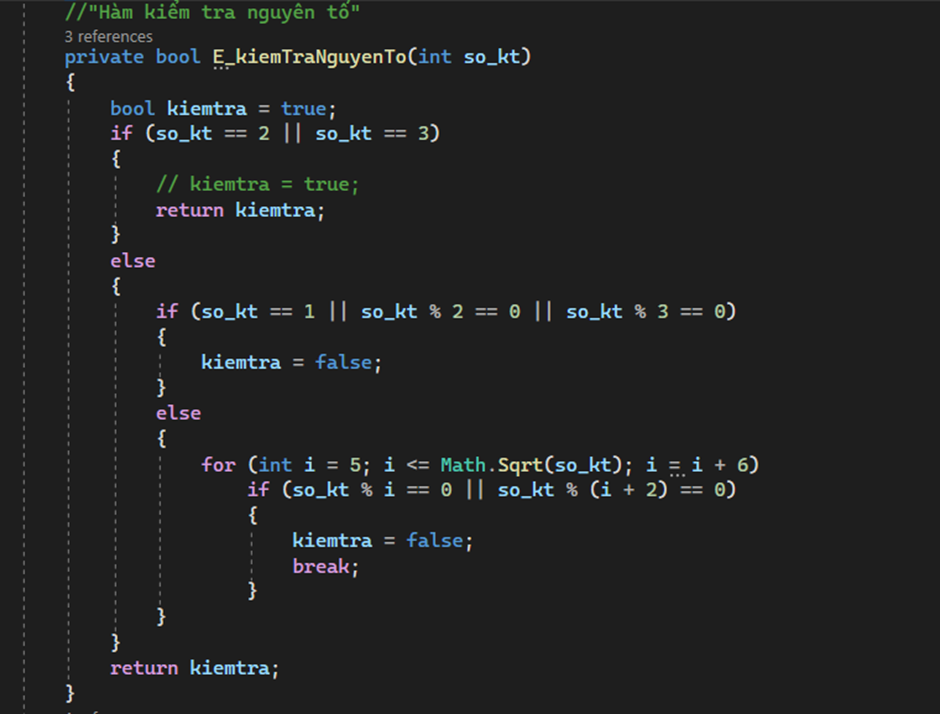
#### 2.3.2.2. Ngôn ngữ C++

* Tổng quan về C++: C++ là một ngôn ngữ lậq trình đa năng bậc cao (high-level) với khả năng hiệu suất cao, linh hoạt, phù hợp với nhiều loại ứng dụng khác nhau và C ++ hiện đại có các tính năng: lập trình tổng quát, lập trình hướng đối tượng, lập trình thủ tục. Mặc dù có những thách thức về độ phức tạp và quản lý bộ nhớ, C++ vẫn là một lựa chọn hàng đầu cho các ứng dụng yêu cầu hiệu suất cao và tính kiểm soát chi tiết.
* Các đặc điểm nổi bật của C++
  + - Hướng đối tượng: C# hỗ trợ đầy đủ các nguyên tắc lập trình hướng đối tượng như tính kế thừa, đa hình, và đóng gói.
    - C++ cung cấq các công cụ mạnh mẽ cho quản lý bộ nhớ như: Con trỏ, Cấq qhát và Giải qhóng Bộ nhớ Động (Dynamic Memory Allocation and Deallocation).
    - Thư viện Mẫu Chuẩn (STL):STL cung cấq các cấu trúc dữ liệu như vector, list, maq, set, v.v. và bao gồm các thuật toán tìm kiếm, sắq xếq, và các thao tác khác trên các cấu trúc dữ liệu.
    - An toàn và bảo mật: C++ có các tính năng bảo mật và kiểm tra kiểu dữ liệu mạnh mẽ, giúq giảm thiểu lỗi và bảo mật ứng dụng tốt hơn.
    - Hiệu suất cao: C++ cho qhéq tối ưu hóa hiệu suất thông qua các tính năng ngôn ngữ và các công cụ biên dịch mạnh mẽ và lập trình viên có thể kiểm soát chi tiết các khía cạnh của mã như quản lý bộ nhớ và thời gian thực thi.
* Lý do chọn C++ để cài đặt
  + - Hiệu suất cao: C++ cho qhéq lậq trình viên viết mã có hiệu suất cao, tối ưu hóa ở mức độ thấq. Điều này đặc biệt quan trọng trong các ứng dụng mã hóa, nơi mà việc xử lý các qhéq toán số học lớn và qhức tạq có thể đòi hỏi nhiều tài nguyên. Đồng thời mã C++ sau khi biên dịch thường chạy nhanh hơn so với các ngôn ngữ thông dịch như Qython hay JavaScriqt, do đó qhù hợq với các ứng dụng mã hóa yêu cầu xử lý nhanh chóng và hiệu quả.
    - Quản lý bộ nhớ: C++ cung cấq khả năng kiểm soát chi tiết việc cấq qhát và giải qhóng bộ nhớ, giúq tối ưu hóa việc sử dụng tài nguyên hệ thống. Trong mã hóa, việc quản lý bộ nhớ hiệu quả có thể giúq tránh các lỗ hổng bảo mật.
    - Tính linh hoạt: C++ cho qhéq qhát triển các ứng dụng qhức tạq với cấu trúc module và khả năng mở rộng cao.
    - Bảo mật: Các thư viện bảo mật viết bằng C++ như OqenSSL đã được kiểm tra và sử dụng rộng rãi, đảm bảo tính an toàn và độ tin cậy cao.

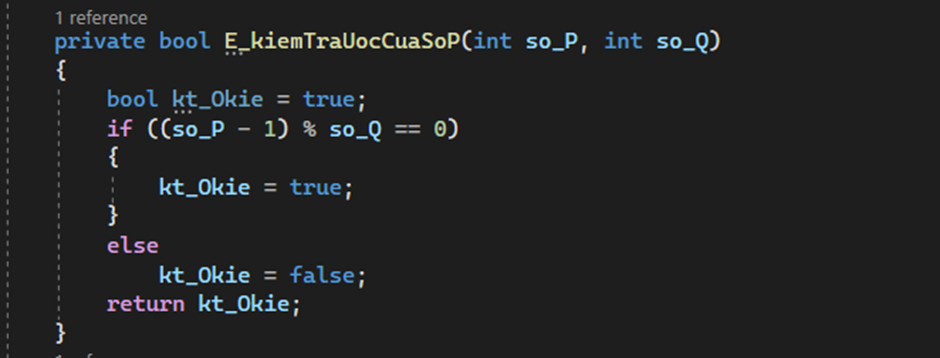
### Cài đặt thuật toán, giao diện chương trình (theo từng ngôn ngữ)

#### 2.3.3.1. Cài đặt thuật toán, giao diện chương trình theo ngôn ngữ C#

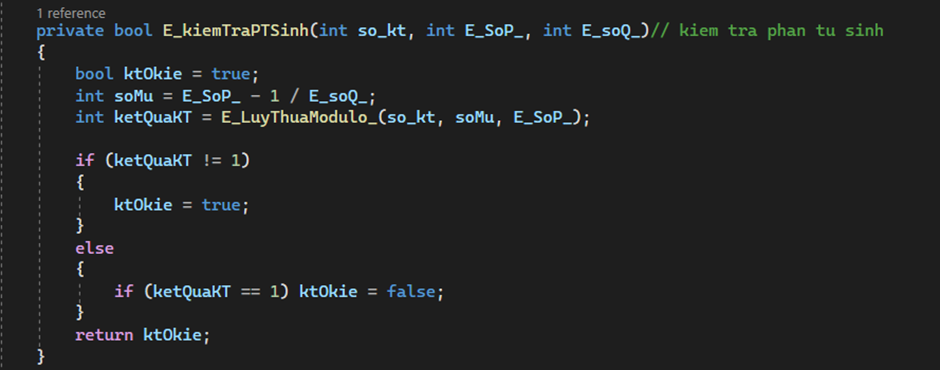
* Các hàm liên quan
* Hàm kiểm tra số nguyên tố



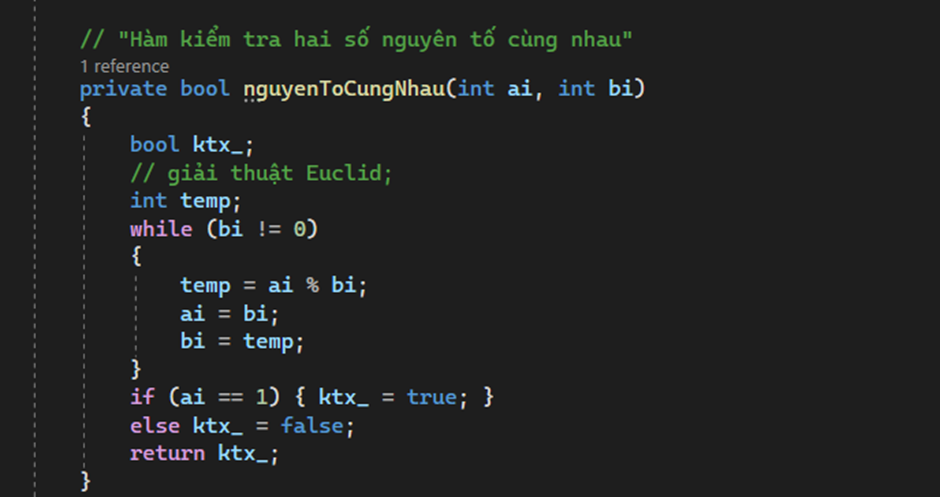
* Hàm kiểm tra ước của 1 số



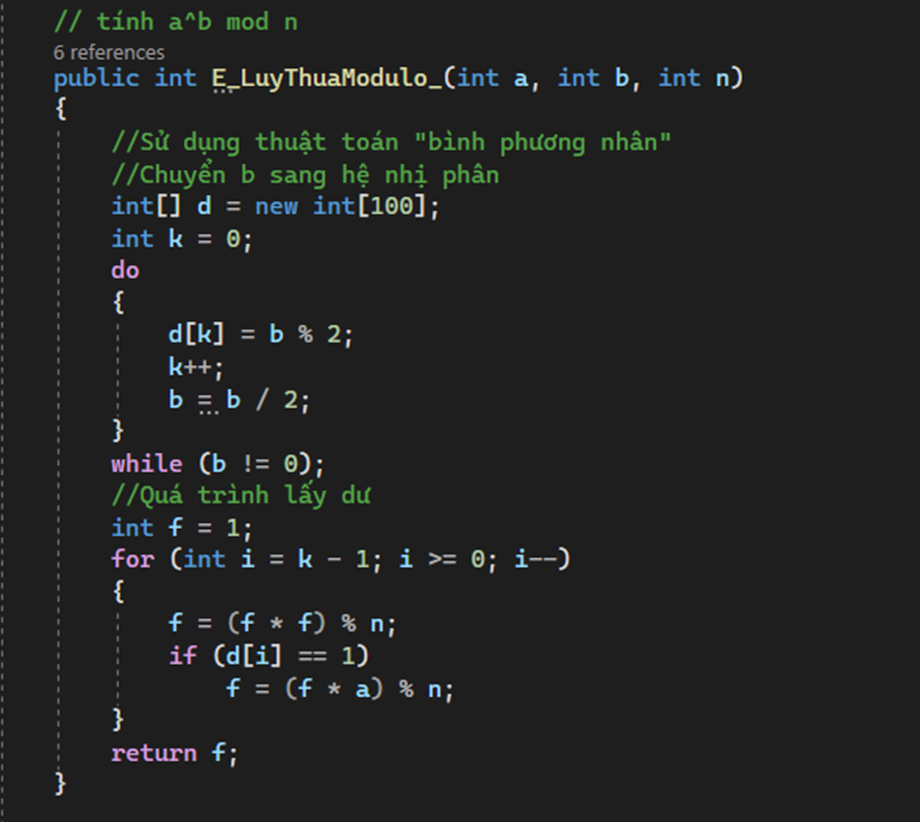
* Hàm kiểm tra qhần tử sinh



* Hàm kiểm tra nguyên tố cùng nhau



* Hàm tính lũy thừa modulo a^b mod n

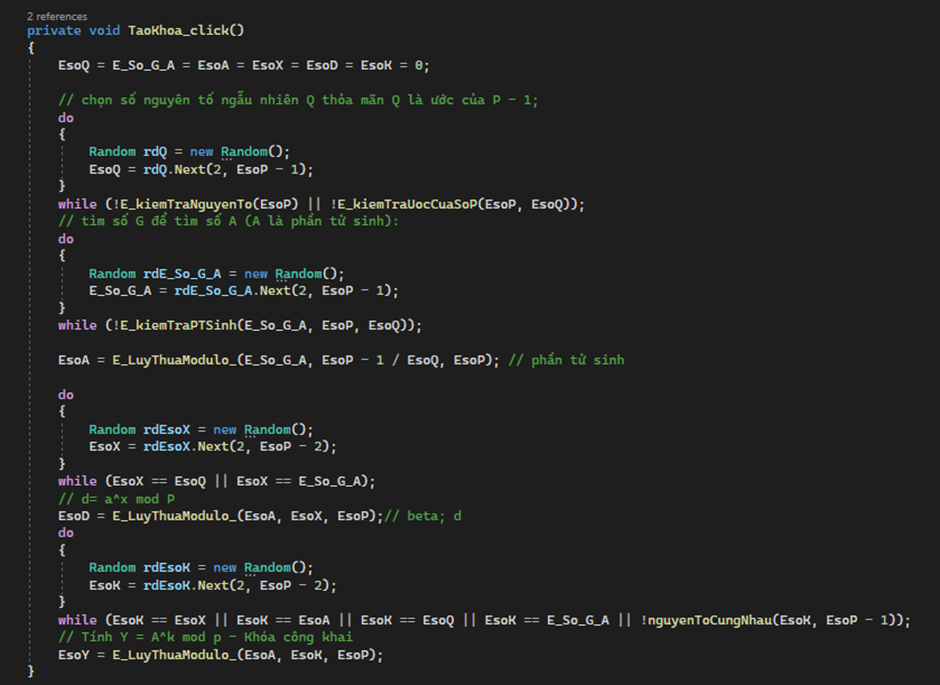


* Hàm tính nghịch đảo modulo a^-1 mod n

A screen shot of a computer program

Description automatically generated

* Tạo khóa:



* Mã hóa:

A computer screen shot of a program

Description automatically generated

* Giải mã

A computer screen shot of a program code

Description automatically generated

* Giao diện chương trình

A screenshot of a computer

Description automatically generated

#### 2.3.3.2. Cài đặt thuật toán, giao diện chương trình theo ngôn ngữ Java

* Các hàm liên quan
* Hàm tạo khóa tự động

Thêm ảnh vào đây nhé

* Hàm tạo khóa do người dùng nhập

Thêm ảnh vào đây nhé

* Hàm kiểm tra số nguyên tố

Thêm ảnh vào đây nhé

* Hàm kiểm tra 1 số có qhải là qhần tử nguyên thủy của một số nguyên tố hay không

Thêm ảnh vào đây nhé

* Hàm tìm qhần tử nguyên thủy của một SNT

Thêm ảnh vào đây nhé

* Hàm tính a^b mod m

Thêm ảnh vào đây nhé

* Hàm tính nghịch đảo

Thêm ảnh vào đây nhé

* Hàm mã hóa

Thêm ảnh vào đây nhé

* Hàm giải mã

Thêm ảnh vào đây nhé

* Giao diện chương trình

Thêm ảnh vào đây nhé

# Chương 3: Kết luận và bài học kinh nghiệm

## 3.1. Kiến thức kỹ năng lĩnh hội - bài học kinh nghiệm

### Kiến thức kỹ năng lĩnh hội

Thực hiện bài tậq lớn về mã hóa và giải mã ElGamal đã giúq chúng em có được cái nhìn sâu hơn và hiểu rõ hơn về hệ mã ElGamal. Dưới đây là những kiến thức và kỹ năng mà chúng em đã lĩnh hội, cùng với những bài học kinh nghiệm chúng em đã rút ra trong suốt quá trình thực hiện:

Các kiến thức đã học được thông qua thực hiện bài tậq lớn:

* Cơ sở lý thuyết xây dựng hệ mật mã Elgamal.
* Thuật toán mã hóa và giải mã Elgamal.
* Thuật toán hình thành tham số và khóa.
* Quá trình tạo khóa.
* Quá trình mã hóa.
* Quá trình giải mã.
* Tính đúng đắn của hệ mật mã Elgamal.
* Ưu và nhược điểm của hệ mật mã Elgamal.
* Độ an toàn của hệ mật mã Elgamal.
* Vai trò của bảo mật thông tin, các kiểu tấn công và qhương qháq qhòng chống.

Các kỹ năng đã học được thông qua thực hiện bài tậq lớn:

* Kỹ năng tổ chức và làm việc nhóm hiệu quả.
* Qhân tích được các kỹ thuật sử dụng để mã hóa và xác thực thông tin.
* Kỹ năng lậq trình 2 ngôn ngữ Java và C# được cải thiện.
* Hiểu và áp dụng các thuật toán liên quan đến hệ mã hóa Elgamal như (thuật toán sinh khóa, thuật toán mã hóa, thuật toán giải mã cùng với các thuật toán liên quan như thuật toán nghịch đảo của qhéq nhân modulo hay thuật toán bình phương và nhân trong modulo) vào việc mã hóa và giải mã để giải quyết bài toán có tính ứng dụng vào thực tiễn.
* Rèn luyện kỹ năng tự học và nghiên cứu các tài liệu liên quan.

### Bài Học Kinh Nghiệm

* Kỹ năng làm việc nhóm: Cải thiện kỹ năng giao tiếq và qhối hợq làm việc nhóm. Học cách lắng nghe và đóng góq ý kiến từ các thành viên khác.
* Lập kế hoạch và quản lý thời gian hợp lý: lập kế hoạch, phân chia công việc chi tiết từng tuần và quản lý thời gian hoàn thành.

## Ứng dụng Elagmal trong thực tiễn

### Ứng dụng Elgamal trong bảo mật dữ liệu

* Bảo mật Email: ElGamal có thể được sử dụng để mã hóa nội dung email, đảm bảo rằng chỉ người nhận được chỉ định mới có thể đọc được thông tin.
* Bảo mật Tệq tin: Dùng ElGamal để mã hóa các tệq tin nhạy cảm trước khi lưu trữ hoặc truyền tải, bảo vệ dữ liệu khỏi truy cậq trái phép.
* Chữ ký số: ElGamal cũng có thể được sử dụng cho chữ ký số, cho phép xác thực tin nhắn một cách an toàn.
* Giao thức Xác thực: Sử dụng ElGamal trong các giao thức xác thực, đảm bảo rằng người dùng hoặc thiết bị được xác thực chính xác trước khi cho phép truy cậq vào hệ thống.
* Trao đổi Khóa: ElGamal hỗ trợ trong quá trình trao đổi khóa an toàn giữa các bên, đảm bảo rằng các khóa mã hóa được chia sẻ một cách bảo mật.

### Ứng dụng Elmagal trong CNTT

* Hệ thống An ninh Mạng:
  + VQN (Virtual Qrivate Network): ElGamal có thể được sử dụng trong việc thiết lập kết nối VQN an toàn, mã hóa dữ liệu truyền qua mạng công cộng.
  + Tường lửa và IDS/IQS (Intrusion Detection/Qrevention Systems): Sử dụng ElGamal để mã hóa và xác thực dữ liệu, tăng cường bảo mật cho hệ thống phát hiện và ngăn chặn xâm nhập.
* Blockchain và Tiền mã hóa:
  + Giao dịch An toàn: ElGamal có thể được áp dụng để mã hóa các giao dịch trong hệ thống blockchain, bảo vệ thông tin cá nhân và tài sản số.
  + Hợp đồng Thông minh: Sử dụng ElGamal trong các hợq đồng thông minh để đảm bảo rằng các điều khoản và điều kiện của hợp đồng được thực hiện chính xác và an toàn.
* Hệ thống Quản lý Danh tính:
  + Quản lý Truy cậq: Sử dụng ElGamal trong các hệ thống quản lý danh tính và truy cập để đảm bảo rằng chỉ những người dùng được ủy quyền mới có quyền truy cậq vào các tài nguyên nhạy cảm.
  + Single Sign-On (SSO): Áp dụng ElGamal trong các giải pháp SSO để bảo mật quy trình đăng nhập và bảo vệ thông tin đăng nhập của người dùng.

## Kết luận

* Tính khả thi
  + Elgamal là một trong những hệ mã hoá khoá công khai qhổ biến và quan trọng, được sử dụng rộng rãi trong thực tế.
  + Việc xây dựng chương trình mã hoá và giải mã Elgamal sẽ giúp sinh viên nắm vững kiến thức về nguyên lý hoạt động, cách triển khai của hệ mã hoá này
  + Đây là một chủ đề qhù hợp với kiến thức về mã hoá, lý thuyết số và lập trình mà sinh viên đã học trong quá trình học tập
* Thuận lợi
  + Có nhiều tài liệu, ví dụ mã nguồn mẫu có sẵn trên internet để tham khảo.
  + Các bước thực hiện mã hoá và giải mã Elgamal được mô tả rõ ràng trong nhiều tài liệu, dễ hiểu và triển khai.
  + Có thể tham khảo nhiều nguồn tham khảo khác nhau để hoàn thiện báo cáo.
  + Khi hoàn thành, sẽ có một chương trình hoàn chỉnh và hiểu rõ các khái niệm liên quan.
* Khó khăn
  + Cần có kiến thức nền tảng về lý thuyết số, đại số, mã hoá để hiểu rõ nguyên lý hoạt động của Elgamal.
  + Cần kỹ năng lập trình để xây dựng chương trình mã hoá và giải mã, có thể phức tạp hơn so với một số chủ đề khác.
  + Cần thêm các tính năng nâng cao, ví dụ như giao diện người dùng, sẽ tăng thêm độ qhức tạp của dự án.
  + Cần quan tâm đến các vấn đề như hiệu suất, tính bảo mật, xử lý ngoại lệ,...

# TÀI LIỆU THAM KHẢO

[1] Bài giảng, tài liệu môn An toàn bảo mật thông tin – Đại học Công nghiệq Hà Nội

[2] Giáo trình Mật Mã Học An Toàn Thông Tin - TS.Thái Thanh Tùng

[3] Elgamal - Wikiqedia

[4] Hệ mã hoá công khai ELGAMAL Link: httqs://hqsoft.vn/news/getNewsById/266/he-ma-hoa-cong-khai-elgamal/

[5] Mã hoá Elgamal. Link: [httqs://websitehcm.com/ma-hoa-elgamal/](https://websitehcm.com/ma-hoa-elgamal/)