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**: **Tìm hiểu về chữ ký tự điện tử Elgamal và viết ứng dụng minh họa (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**](#_gjdgxs)

[**Chương 1: Tổng quan về đề tài nghiên cứu 4**](#_30j0zll)

[1.1.](#_1fob9te) Tổng quan về an toàn và bảo mật thông tin 4

[1.2.](#_3znysh7) Lý do chọn đề tài 5

[1.3.](#_2et92p0) Nội dung nghiên cứu 6

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

[**Chương 2: Kết quả nghiên cứu 8**](#_3dy6vkm)

[2.1.](#_1t3h5sf) Nghiên cứu, tìm hiểu hệ mã hóa khóa công khai 8

[2.1.1.](#_4d34og8) Ý tưởng của Whitfield Diffie và Martin Hellman 8

[2.1.2.](#_2s8eyo1) Cấu trúc của một hệ mã PKC 9

[2.1.3.](#_17dp8vu) Độ an toàn của mã hoá khoá công khai 9

[2.1.4.](#_3rdcrjn) Nhận xét về hệ mã hoá khoá công khai 10

[2.2.](#_26in1rg) Nghiên cứu, tìm hiểu về thuật toán Elgamal 10

[2.2.1.](#_lnxbz9) Giới thiệu chung về hệ mã Elgamal 10

[2.2.2.](#_35nkun2) Mã hoá và giải mã hệ mã Elgamal 10

[2.2.3.](#_1ksv4uv) Độ an toàn 12

[2.2.4.](#_44sinio) . Ưu và nhược điểm của hệ mã Elgamal 12

[2.2.5.](#_2jxsxqh) Thuật toán Elgamal 13

[2.3.](#_z337ya) Thiết kế chương trình, cài đặt thuật toán. 14

[2.3.1.](#_3j2qqm3) Kịch bản chương trình: 14

[2.3.2.](#_1y810tw) Giới thiệu ngôn ngữ lập trình sử dụng để cài đặt thuật toán. 15

[2.3.3.](#_4i7ojhp) Cài đặt thuật toán, giao diện chương trình (theo từng ngôn ngữ) 17

[**Chương 3: Kết luận và bài học kinh nghiệm 32**](#_2xcytpi)

[2.4.](#_1ci93xb) 3.1. Kiến thức kỹ năng lĩnh hội - bài học kinh nghiệm 32

[3.1.1.](#_3whwml4) Kiến thức kỹ năng lĩnh hội 32

[3.1.2.](#_2bn6wsx) Bài Học Kinh Nghiệm 33

[3.2.](#_qsh70q) Ứng dụng Elagmal trong thực tiễn 33

[3.2.1.](#_3as4poj) Ứng dụng Elgamal trong bảo mật dữ liệu 33

[3.2.2.](#_1pxezwc) Ứng dụng Elmagal trong CNTT 33

[3.3.](#_49x2ik5) Kết luận 34

[**TÀI LIỆU THAM KHẢO 36**](#_2p2csry)

# 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 ElGamal là một thuật toán mã hóa khóa công khai được sử dụng rộng rãi trong lĩnh vực an toàn và bảo mật thông tin, đặc biệt trong việc tạo chữ ký điện tử. Để hiểu rõ hơn về cách hoạt động của chữ ký ElGamal, trước hết chúng ta cần hiểu về an toàn và bảo mật thông tin, cũng như các vấn đề liên quan đến nó.

Thứ nhất, an toàn và bảo mật thông tin là một lĩnh vực quan trọng và thiết yếu trong kỹ thuật và công nghệ hiện đại. Nó liên quan đến việc bảo vệ thông tin quan trọng khỏi sự truy cập trái phép, sửa đổi hoặc phá hoại bởi những đối tượng không có quyền truy cập. Trong thời đại công nghệ thông tin phát triển mạnh mẽ, bảo mật thông tin đóng vai trò rất lớn trong việc bảo vệ tài nguyên số của các cá nhân và tổ chức.

Với sự phát triển nhanh chóng của công nghệ thông tin, các hệ thống truyền thông dữ liệu trở nên phổ biến hơn bao giờ hết, khiến vấn đề an toàn và bảo mật thông tin càng trở nên cấp bách.

Các vấn đề liên quan đến bảo mật thông tin bao gồm:

1. Mã hóa: Mã hóa là quá trình chuyển đổi dữ liệu thành một định dạng khác, bảo vệ thông tin khỏi việc bị truy cập trái phép. Các thuật toán mã hóa phổ biến như DES, AES, RSA, ElGamal...
2. Chữ ký số: Là cơ chế xác thực tài liệu, đảm bảo tính toàn vẹn và nguồn gốc của thông tin. Chữ ký số thường dựa trên một thuật toán mã hóa và khóa bí mật.
3. Quản lý khóa: Quy trình quản lý các khóa mã hóa và giải mã, đảm bảo rằng chỉ những người có quyền mới có thể truy cập vào thông tin.
4. Quản lý danh tính: Xác thực danh tính người dùng và bảo vệ thông tin cá nhân khỏi việc truy cập trái phép.
5. Kiểm thử bảo mật: Kiểm tra hệ thống để phát hiện các lỗ hổng bảo mật, đảm bảo hệ thống đáp ứng yêu cầu an toàn thông tin.

Thứ hai, Thuật toán ElGamal là một thuật toán mã hóa khóa công khai, được sử dụng rộng rãi trong bảo mật thông tin và chữ ký số. Đặc biệt, nó sử dụng bài toán logarit rời rạc để tạo ra một hệ thống bảo mật mạnh mẽ.

Chữ ký ElGamal sử dụng cặp khóa công khai và khóa bí mật. Quá trình tạo chữ ký gồm ba bước chính:

1. Tạo khóa: Bao gồm tạo khóa công khai và khóa bí mật.
2. Tạo chữ ký: Tạo chữ ký dựa trên thông tin tài liệu cần ký và khóa bí mật.
3. Kiểm tra chữ ký: Kiểm tra tính hợp lệ của chữ ký bằng khóa công khai.

## 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 12 chúng em lựa chọn đề tài “***Tìm hiểu về chữ ký điện tử Elgamal và viết ứng dụng minh họa***” 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 vô cùng quan trọng. Với sự phát triển nhanh chóng của công nghệ thông tin và việc giao dịch, truyền tải dữ liệu qua Internet ngày càng trở nên phổ biến, việc bảo vệ thông tin cá nhân và tổ chức trở thành vấn đề cấp thiết. Hiểu và xây dựng chương trình chữ ký điện tử ElGamal giúp chúng em trang bị kiến thức và kỹ năng cần thiết trong lĩnh vực bảo mật, đồng thời cũng nâng cao khả năng áp dụng các giải pháp bảo mật trong thực tế.

2. ElGamal là một trong những thuật toán bảo mật phổ biến

ElGamal là một thuật toán mã hóa khóa công khai, được sử dụng rộng rãi trong việc tạo chữ ký số và mã hóa dữ liệu. Đây là một thuật toán có tính bảo mật cao, dựa trên bài toán **logarit rời rạc**, và là một phần quan trọng trong các giao thức bảo mật hiện nay như SSL/TLS, chữ ký số, và giao dịch trực tuyến. Việc hiểu rõ và xây dựng chương trình chữ ký điện tử ElGamal giúp chúng em hiểu sâu hơn về cách thức hoạt động của thuật toán này, từ đó áp dụng vào các ứng dụng bảo mật thực tế.

3. Phát triển kỹ năng lập trình

Xây dựng chương trình chữ ký điện tử ElGamal giúp chúng em cải thiện kỹ năng lập trình, đặc biệt là trong việc xử lý các số nguyên lớn và các thuật toán toán học liên quan. Thuật toán ElGamal yêu cầu chúng em phải làm quen với các phép toán mô-đun, logarit rời rạc và các kỹ thuật xử lý số học nâng cao. Quá trình xây dựng chương trình cũng giúp bọn em nâng cao khả năng lập trình, từ việc thiết kế thuật toán đến việc cài đặt và kiểm thử.

4. Ứng dụng thực tế

Chữ ký điện tử ElGamal không chỉ là một thuật toán mã hóa lý thuyết, mà còn có nhiều ứng dụng thực tế trong bảo mật thông tin. Các ứng dụng phổ biến của chữ ký ElGamal bao gồm: bảo mật email, giao dịch trực tuyến, chữ ký số trong các tài liệu điện tử, và xác thực thông tin trong các hệ thống phân tán. Việc xây dựng chương trình chữ ký ElGamal giúp chúng em hiểu rõ hơn về cách thuật toán này được áp dụng trong thực tế, và làm thế nào để bảo vệ thông tin trong môi trường mạng.

Tóm lại, Việc xây dựng chương trình chữ ký điện tử ElGamal không chỉ là một đề tài nghiên cứu thú vị, mà còn mang lại cho chúng em rất nhiều lợi ích. Qua đó, chúng em không chỉ phát triển kỹ năng lập trình mà còn hiểu rõ hơn về cách bảo mật thông tin trong thời đại số. Điều này không chỉ giúp chúng em nắm vững lý thuyết về bảo mật mà còn có thể áp dụng vào các ứng dụng thực tế trong công việc và cuộc sống hàng ngày.

## Nội dung nghiên cứu

Trong báo cáo thực nghiệm “***Tìm hiểu về chữ ký điện tử Elgamal và viết ứng dụng minh họa***”, nhóm 12 chúng em cùng nhau nghiên cứu về các nội dung:

1. *Tìm hiểu về chữ ký điện tử*

* Giới thiệu về chữ ký điện tử
* Khái niệm thế nào là chữ ký điện tử
* Ứng dụng của chữ ký điện tử
* Tầm quan trọng
* Một số ưu nhược điểm của chữ ký điện tử.

1. *Chữ ký điện tử ElGamal*

* Lược đồ chữ ký điện tử ElGamal.
* Ví dụ minh hoạ.
* Độ an toàn của chữ ký điện tử ElGamal

1. *Tìm hiểu phương pháp mã hóa bất đối xứng ứng dụng trong chữ ký điện tử*

* Mã hóa bất đối xứng là gì
* Đặc điểm
* Ưu điểm
* Hạn chế
* Ứng dụng trong chữ ký số
* Chữ ký số
* Chữ ký số sử dụng hệ mật mã Elgamal
* Ưu điểm
* Ý nghĩa

1. Tìm hiểu về hàm băm SHA

* Giới thiệu hàm băm Hash
* Tính chất cơ bản của hàm băm Hash
* Danh sách các hàm băm mật mã học
* Ứng dụng hàm băm Hash
* Thuật toán hàm băm SHA-1
* Giới thiệu hàm băm SHA-1
* Thuật toán băm SHA-1
* Thuật toán hàm băm SHA-256
* Mã hoá SHA-256
* Ứng dụng của SHA-256

Áp dụng thực hiện xây dựng chương trình với các ngôn ngữ C#, Java .

## 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 java, 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, alpha là phần tử nguyên thuỷ của Z\*q
  + Bài toán logarithm rời rạc có thể được phát 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 phương pháp vét cạn (tức là duyệt tất cả phần tử a) để tìm a thoả mãn. Khi đó độ phức tạp là sẽ là O(q). Vấn đề đặt ra là nêu q lớn, rất lớn thì để thực hiện phương pháp 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 Euclid mở rộng tìm phần tử nghịch đảo*

Cho 2 số nguyên r0, r1 tìm r1-1 theo mod r0

**Intput**: r0, r1

**Output** : r1-1 theo mod r0 (Nếu tồn tại)

* Dùng thuật toán Euclide mở rộng để tìm các số nguyên s và t sao cho s. r0 +t. r1 = gcd(r0, r1) =d
* Nếu d>1 thì r1-1 mod r0 không tồn tại. Ngược lại nếu d=1 thì return(t)

**Để tìm được s, t ta dùng công thức sau :**

s0 =1, t0 =0

s0 =0, t0 =1

si = s(i-2) – q(i-1)\* s(i-1)

ti =  t(i-2) – q(i-1)\* t(i-1)

Trong đó: Với i=0,1,2,3,..

ri =qi+1\*ri+1 + ri+2

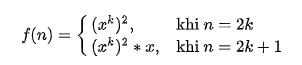
Thuậttoán dừng lại khi phần dư ri+2 =0

*Thuật toán: Bình phương và nhân*

Công thức đệ quy: để tính luỹ thừa tự nhiên bậc n của x thực hiện như sau:

Với n=0 thì xn =1

Với n>0 ta có công thức bình phương và nhân:



Như vậy phép tính xn được đệ quy về một số phép bình phương và phép nhân

*Thuật toán: Sơ đồ chữ ký điện tử Elgamal*

Sơ đồ chữ ký Elgamal là được viện tiêu chuẩn và công nghệ quốc gia Mỹ sửa đổi thành chuẩn chữ ký số. Sơ đồ chữ ký Elgamal không nhất thiết phải giống như hệ thống mã hóa công khai Elgamal. Điều này có nghĩa là có nhiều chữ ký hợp lệ cho cùng một thông điệp bất kỳ. Thuật toán xác minh phải có khả năng chấp nhận bất kỳ chữ ký hợp lệ nào khi xác minh.

Sơ đồ Elgamal được định nghĩa như sau:

* **Tạo cặp khóa (bí mật, công khai) (a, k) :**

+ Chọn phần tử nguyên tử **α ϵ Zp\***. Đặt **P = Zp\***,  **A = Zp\* x Zp-1**

+ Chọn khoá bí mật là **α ϵ Zp\***. Tính khoá công khai **β ≡ αa mod p**.

+ Định nghĩa tập khoá: **={(p, α, a, β) : β ≡ αa mod p}.**

+ Các giá trị **p, α, β** được công khai, phải giữ bí mật **a.**

* **Ký số**

+ Dùng 2 khóa ký: khoá **a** và số ngẫu nhiên **k ϵ Zp-1\***

+ Vì **k ϵ Zp-1\***, nên nguyên tố cùng p-1, do đó tồn tại k-1 mod (p-1)

+ Chữ ký trên **x** **ϵ P là y = sigk(x, k) = (γ, δ), y ϵ A**

Trong đó **γ ϵ Zp\*, δ ϵ Zp-1\*:**

**γ = αk mod p** và

**δ= (x-a\* γ) \*k-1 mod (p-1)**

* **Kiểm tra chữ ký**

**verk (x, γ, δ)= TRUE ⬄ βγ \* γδ ≡ αx mod p**

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

### Kịch bản chương trình:

* Bước 1:Tại giao diện tạo khoá chúng ta cần nhấn vào phần tạo khoá để sinh ra khoá công khai và khoá bí mật
* Bước 2: Tại giao diện văn bản ta lần lượt thực hiện các bước sau:
* Tải văn bản cần ký
* Ký vào văn bản đó
* Sau khi đã ký xong thì lưu lại chữ ký
* Bước 3: Tại giao diện xác nhận văn bản ta thực hiện các bước
* Tải văn bản đã ký và cần xác nhận lên.
* Tải chữ ký kèm theo đã được ký ở văn bản
* Xác nhận chữ ký
* Kiểm tra sự toàn vẹn của tài liệu/văn bản
* Nếu văn bản đã được chỉnh sửa hoặc chữ ký kèm theo không chính xác sẽ xuất ra thông báo là văn bản đã được chỉnh sửa hoặc chữ ký không chính xác.
* Nếu văn bản và chữ ký đều chính xác thì chương trình sẽ thông báo chữ ký đã chính xác.

### Giới thiệu ngôn ngữ lập 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 hợp 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 cập 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.
    - Lập trình bất đồng bộ: Hỗ trợ mạnh mẽ cho lập 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, phù 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 cấp các thư viện toán học và mã hóa mạnh mẽ, giúp 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, giúp 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 phát triển tích hợp (IDE) mạnh mẽ, giúp việc lập trình và gỡ lỗi trở nên dễ dàng hơn.

#### 2.3.2.2. Ngôn ngữ java

* Tổng quan về java: java là một ngôn ngữ lập 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 java 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, java 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 java
  + - Hướng đối tượng: java hỗ trợ đầy đủ các nguyên tắc lập trình hướng đối tượng như tính kế thừa, đa hình, và đóng gói.
    - C++ cung cấp các công cụ mạnh mẽ cho quản lý bộ nhớ như: Con trỏ, Cấp phát và Giải phóng Bộ nhớ Động (Dynamic Memory Allocation and Deallocation).
    - Thư viện Mẫu Chuẩn (STL):STL cung cấp các cấu trúc dữ liệu như vector, list, maq, set, v.v. và bao gồm các thuật toán tìm kiếm, sắp xếp, và các thao tác khác trên các cấu trúc dữ liệu.
    - An toàn và bảo mật: Java 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.
    - Hiệu suất cao: Java 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 Java để cài đặt
  + - Hiệu suất cao: Java cho phép lập trình viên viết mã có hiệu suất cao, tối ưu hóa ở mức độ thấp. Điều này đặc biệt quan trọng trong các ứng dụng mã hóa, nơi mà việc xử lý các phép toán số học lớn và phức tạp 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 python hay JavaScript, do đó qhù hợq với các ứng dụng mã hóa yêu cầu xử lý nhanh chóng và hiệu quả.
    - Quản lý bộ nhớ: Java cung cấp khả năng kiểm soát chi tiết việc cấp phát và giải phóng bộ nhớ, giúp tối ưu hóa việc sử dụng tài nguyên hệ thống. Trong mã hóa, việc quản lý bộ nhớ hiệu quả có thể giúp tránh các lỗ hổng bảo mật.
    - Tính linh hoạt: Java cho phép phát triển các ứng dụng phức tạp với cấu trúc module và khả năng mở rộng cao.
    - Bảo mật: Các thư viện bảo mật viết bằng Java như OpenSSL 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
* Tạo cặp khóa công khai và khóa bí mật
* Mã hóa thông điệp với khóa công khai
* Giải mã thông điệp
* Tạo chữ kỹ cho thông điệp
* Kiểm tra tính hợp lệ của chữ ký
* Giao diện chương trình

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

* Các hàm liên quan
* Tạo cặp khóa công khai và khóa bí mật
* Mã hóa thông điệp với khóa công khai
* Giải mã thông điệp
* Tạo chữ kỹ cho thông điệp
* Kiểm tra tính hợp lệ của chữ ký
* Giao diện chương trình

# 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 tập lớn về mã hóa và giải mã ElGamal đã giúp 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.
* Phân tích được các kỹ thuật sử dụng để mã hóa và xác thực thông tin.
* Kỹ năng lập 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 phép 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 tiếp và phối hợp làm việc nhóm. Học cách lắng nghe và đóng góp 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 Elgamal 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 Tệp tin: Dùng ElGamal để mã hóa các tệp tin nhạy cảm trước khi lưu trữ hoặc truyền tải, bảo vệ dữ liệu khỏi truy cập 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 cập 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 Private Network): ElGamal có thể được sử dụng trong việc thiết lập kết nối VPN an toàn, mã hóa dữ liệu truyền qua mạng công cộng.
  + Tường lửa và IDS/IQS (Intrusion Detection/Prevention 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 hợp 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 cập: 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 cập 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 phổ 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 phù 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 phức 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/)