

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO THỰC HÀNH MÔN
CẤU TRÚC RỜI RẠC

Người hướng dẫn: **GV. NGUYỄN QUỐC BÌNH**

Người thực hiện: **ĐINH PHƯƠNG MY – 52100703**

Lớp: **21050401**

Khoá: **25**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO THỰC HÀNH MÔN
CẤU TRÚC RỜI RẠC

Người hướng dẫn: **GV. NGUYỄN QUỐC BÌNH**

Người thực hiện: **ĐINH PHƯƠNG MY – 52100703**

Lớp: **21050401**

Khoá: **25**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

LỜI CẢM ƠN

Lời nói đầu tiên, em xin được gửi lời cảm ơn chân thành đến toàn bộ giảng viên Trường Đại học Tôn Đức Thắng nói chung cũng như toàn bộ giảng viên Khoa Công nghệ thông tin nói riêng vì đã tạo điều kiện cho em được học bộ môn Cấu trúc rời rạc. Và đặc biệt em gửi lời cảm ơn chân thành nhất đến thầy Nguyễn Quốc Bình – giảng viên giảng dạy và hướng dẫn cho đề tài môn môn Cấu trúc rời rạc của em. Trong suốt quá trình học tập và thực hiện bài báo cáo, thầy luôn giúp đỡ, chỉ bảo tận tình để em có thể tìm ra cách giải quyết những vướng mắc gặp phải và hoàn thiện đề tài này một cách tốt nhất. Lượng kiến thức này chúng em sẽ làm hành trang để áp dụng vào công việc sau này. Không thể nói gì hơn nữa, một lần nữa, bằng cả tấm lòng, em xin chân thành gửi lời cảm ơn sâu sắc đến thầy - người đã dìu dắt lớp trong suốt chặng đường vừa qua!

BÁO CÁO ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Em xin cam đoan đây là công trình nghiên cứu của riêng em và được sự hướng dẫn khoa học của thầy Nguyễn Quốc Bình. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong báo cáo còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào em xin hoàn toàn chịu trách nhiệm về nội dung báo cáo môn học của mình. Trường Đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày 10 tháng 04 năm 2023

Sinh viên thực hiện

(ký tên và ghi rõ họ tên)



Đinh Phương My

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần xác nhận của GV hướng dẫn

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

Phần đánh giá của GV chấm bài

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

TÓM TẮT

Trong bài báo cáo này, em sẽ phân tích và triển khai hệ thống mật mã RSA, một trong những phương pháp mật mã học công khai phổ biến nhất, bằng cách sử dụng số học mô-đun. Bài báo cáo sẽ được chia thành hai phần lớn sau đây:

- Phần 1: Tìm một Modulo nghịch đảo

Trong phần này, em sẽ tìm hiểu cách tìm một Modulo nghịch đảo, một bước quan trọng trong quá trình mã hóa và giải mã RSA. Em sẽ đi sâu vào nguyên tắc và thuật toán của việc tìm Modulo nghịch đảo bằng cách sử dụng thuật toán Euclid mở rộng và định lý Euler. Sau đó em sẽ thực hiện một số ví dụ minh họa để làm rõ hơn về quá trình này.

- Phần 2: Hệ thống mật mã RSA

Trong phần này, em sẽ tìm hiểu về hệ thống mật mã RSA, một hệ thống mật mã công khai được phát minh bởi Ron Rivest, Adi Shamir và Leonard Adleman vào năm 1978. Em sẽ đi sâu vào phương pháp hoạt động của RSA, bao gồm quá trình sinh khóa, quá trình mã hóa và giải mã. Em sẽ tìm hiểu về các thuật toán và công thức tính toán cụ thể để triển khai hệ thống mật mã RSA và cũng sẽ đề cập đến những ưu điểm và nhược điểm của RSA, cùng với các ứng dụng thực tế của nó trong việc bảo vệ thông tin.

MỤC LỤC

LỜI CẢM ƠN	i
PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN	iii
TÓM TẮT	iv
MỤC LỤC	1
DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ	3
CHƯƠNG 1 – MODULO NGHỊCH ĐẢO.....	4
1.1 Lý thuyết về modulo nghịch đảo	4
1.1.1 Định nghĩa.....	4
1.1.2 Tính chất	4
1.2 Thuật toán Euclid mở rộng	4
1.2.1. Khái niệm.....	4
1.2.2 Thuật toán	4
1.3 Thực hành.....	5
1.4 Kết quả đạt được	5
CHƯƠNG 2 – HỆ THỐNG MẬT MÃ RSA.....	7
2.1 Lý thuyết	7
2.1.1 Số nguyên tố	7
2.1.2 Số học mô đun	7
2.1.3 Phân tích thừa số nguyên tố	8
2.1.4 Mật mã RSA	8
2.1.5 Nguyên tắc hoạt động	8
2.1.6 Ưu điểm và nhược điểm.....	9
2.1.6.1 Ưu điểm	9
2.1.6.2 Nhược điểm.....	10
2.2 Thực hành.....	10
2.3 Kết quả đạt được	11

2.4 Phân tích hiệu quả và tính bảo mật	12
2.4.1 Hiệu quả	12
2.4.2 Tính bảo mật	13
2.5 Kết luận	14
TÀI LIỆU THAM KHẢO	15

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

DANH MỤC HÌNH

Hình 1.2 Thuật toán xây dựng phép modulo nghịch đảo.....	5
Hình 1.3.1 Kết quả phép modulo nghịch đảo	5
Hình 1.3.2 Kết quả không tồn tại phép modulo nghịch đảo	6
Hình 2.1.2 Chiếc đồng hồ với mô đun bằng 12	7
Hình 2.2.1 Thuật toán xây dựng hệ thống mật mã RSA ở bước 1, 2, 3, 4.....	11
Hình 2.2.2 Thuật toán xây dựng hệ thống mật mã RSA ở bước 5, 6.....	11
Hình 2.3 Kết quả mã hóa và giải mã bằng hệ thống mật mã RSA thành công.....	12

DANH MỤC BẢNG

CHƯƠNG 1 – MODULO NGHỊCH ĐẢO

1.1 Lý thuyết về modulo nghịch đảo

Trong toán học, modulo nghịch đảo là khái niệm liên quan đến việc tìm một số nguyên sao cho tích của số đó và một số khác với một số modulo cho trước đạt giá trị là 1. Đây là một khái niệm quan trọng trong các thuật toán mã hóa và giải mã, bao gồm cả hệ thống mật mã RSA

1.1.1 Định nghĩa

Giả sử a, n là hai số nguyên và $n > 1$.

Nếu tồn tại một số nguyên b sao cho $(a \times b) \bmod n = 1$ thì b được gọi là modulo nghịch đảo của a , ký hiệu là $a^{-1} \bmod n$.

1.1.2 Tính chất

- Modulo nghịch đảo tồn tại duy nhất cho mỗi số nguyên a , nếu tồn tại.
- Modulo nghịch đảo của $a \bmod n$ tồn tại nếu và chỉ nếu a và n là hai số nguyên cùng nhau ($USCL(a, n) = 1$).
- Modulo nghịch đảo của $a \bmod n$ có thể được tìm thấy bằng cách sử dụng thuật toán Euclid mở rộng hoặc định lý Euler.

1.2 Thuật toán Euclid mở rộng

1.2.1. Khái niệm

Thuật toán mở rộng Euclide là một phương pháp để tìm nghịch đảo của một số nguyên modulo n . Đây là một thuật toán hiệu quả được sử dụng trong đại số đồng dư để tìm nghịch đảo của một số trong giá trị modulo miền.

1.2.2 Thuật toán

Thuật toán Euclide mở rộng có thể được thực hiện dưới dạng một vòng lặp, với các bước thực hiện như sau:

- Bước 1: Khởi tạo các biến n_0, x, y với các giá trị tương ứng là $n, 0, 1$.
- Bước 2: Sử dụng vòng lặp để tính toán giá trị mới cho đến khi $a > 1$.

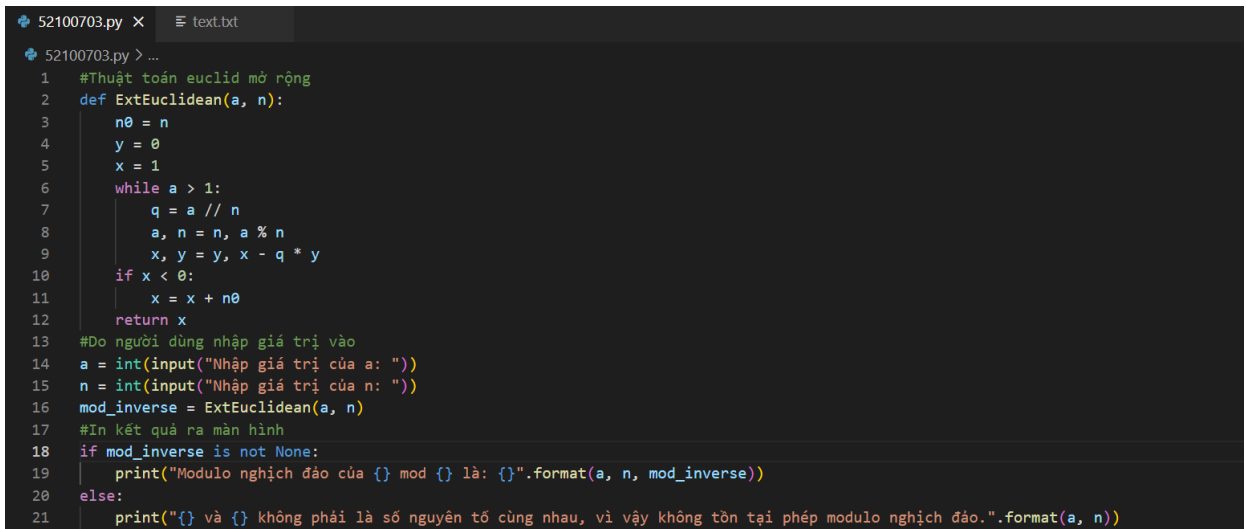
- Bước 3: Trong mỗi bước lặp, tính toán q (thương hiệu của a cho n),

$$a = n, \quad n = a \% m,$$

$$x = y, \quad y = x - q \times y$$
- Bước 4: Sau khi vòng lặp kết thúc, nếu $x < 0$, thì giá trị x là nghịch đảo của một modulo n_0 , ngược lại không tồn tại nghịch đảo của một modulo n_0 .

1.3 Thực hành

Em sẽ thực hiện một chương trình Python để tìm Modulo n nghịch đảo bằng cách sử dụng thuật toán Euclide mở rộng với các bước đã nêu ở phần 1.2.2.



```

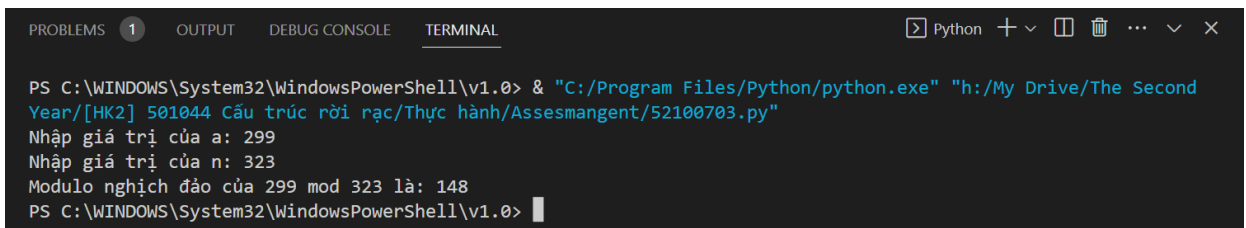
52100703.py x text.txt
52100703.py > ...
1  #Thuật toán euclid mở rộng
2  def ExtEuclidean(a, n):
3      n0 = n
4      y = 0
5      x = 1
6      while a > 1:
7          q = a // n
8          a, n = n, a % n
9          x, y = y, x - q * y
10     if x < 0:
11         x = x + n0
12     return x
13 #Do người dùng nhập giá trị vào
14 a = int(input("Nhập giá trị của a: "))
15 n = int(input("Nhập giá trị của n: "))
16 mod_inverse = ExtEuclidean(a, n)
17 #In kết quả ra màn hình
18 if mod_inverse is not None:
19     print("Modulo nghịch đảo của {} mod {} là: {}".format(a, n, mod_inverse))
20 else:
21     print("{} và {} không phải là số nguyên tố cùng nhau, vì vậy không tồn tại phép modulo nghịch đảo.".format(a, n))

```

Hình 1.2 Thuật toán xây dựng phép modulo nghịch đảo

1.4 Kết quả đạt được

Với dữ liệu nhập vào từ bàn phím $a = 299$ và $n = 323$ thì kết quả khi in ra màn hình là “Modulo nghịch đảo của 299 mod 323 là: 148”



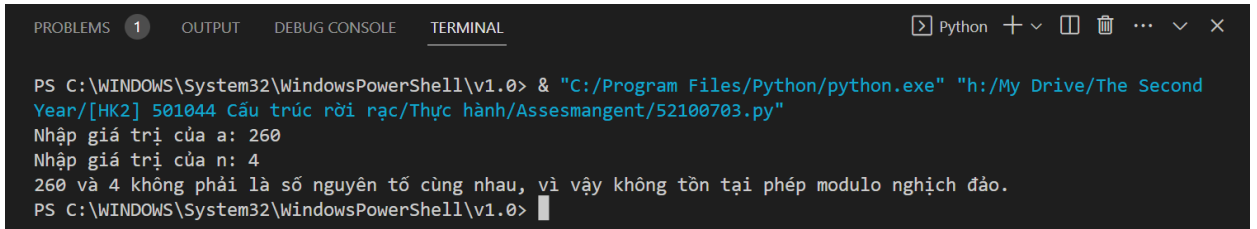
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL
Python + - [ ] [X] ... X
PS C:\WINDOWS\System32\WindowsPowerShell\v1.0> & "C:/Program Files/Python/python.exe" "h:/My Drive/The Second Year/[HK2] 501044 Cấu trúc rời rạc/Thực hành/Assesment/52100703.py"
Nhập giá trị của a: 299
Nhập giá trị của n: 323
Modulo nghịch đảo của 299 mod 323 là: 148
PS C:\WINDOWS\System32\WindowsPowerShell\v1.0>

```

Hình 1.3.1 Kết quả phép modulo nghịch đảo

Với dữ liệu nhập vào từ bàn phím $a = 260$ và $n = 4$ thì kết quả khi in ra màn hình là “260 và 4 không phải là số nguyên tố cùng nhau, vì vậy không tồn tại phép modulo nghịch đảo.”



```
PS C:\WINDOWS\System32\WindowsPowerShell\v1.0> & "C:/Program Files/Python/python.exe" "h:/My Drive/The Second Year/[HK2] 501044 Cấu trúc rời rạc/Thực hành/Assesment/52100703.py"
Nhập giá trị của a: 260
Nhập giá trị của n: 4
260 và 4 không phải là số nguyên tố cùng nhau, vì vậy không tồn tại phép modulo nghịch đảo.
PS C:\WINDOWS\System32\WindowsPowerShell\v1.0>
```

Hình 1.3.2 Kết quả không tồn tại phép modulo nghịch đảo

CHƯƠNG 2 – HỆ THỐNG MẬT MÃ RSA

2.1 Lý thuyết

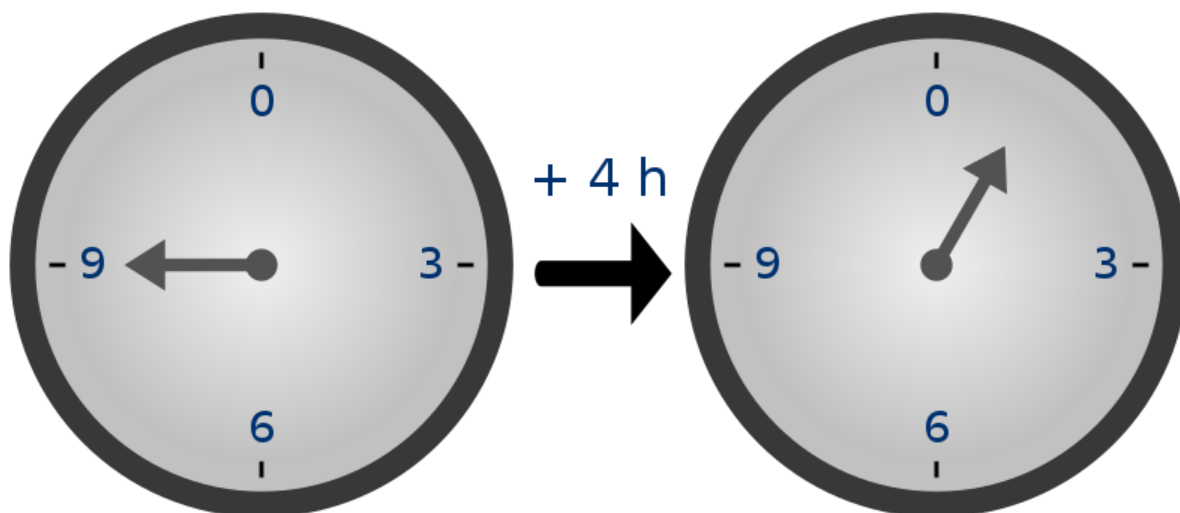
2.1.1 Số nguyên tố

Số nguyên tố là tập hợp các số tự nhiên lớn hơn 1, chia hết cho 1 và chính nó. Hiểu đơn giản hơn, số nguyên tố là những số chỉ có đúng hai ước số là 1 và chính nó.

Trong RSA, cặp khóa bao gồm một khóa công khai và một khóa riêng tư, dựa trên hai số nguyên tố lớn. Việc tạo số nguyên tố là bước quan trọng để đảm bảo tính bảo mật của hệ thống mật mã RSA. Ví dụ, chúng ta có thể tạo hai số nguyên tố lớn là $p = 61$ và $q = 53$.

2.1.2 Số học mô đun

Trong toán học, số học mô đun là một hệ thống số học dành cho số nguyên. Trong số học mô đun, các con số được viết bao quanh lấy nhau thành nhiều vòng tròn cho đến khi chạm đến giá trị đích, gọi là mô đun (tiếng Anh: modulus, số nhiều moduli). Bộ môn nghiên cứu số học mô đun hiện đại được nhà toán học người Đức, Carl Friedrich Gauss phát triển trong cuốn sách của ông có tên Disquisitiones Arithmeticae, xuất bản năm 1801.



Hình 2.1.2 Chiếc đồng hồ với mô đun bằng 12 (Nguồn: [Số học mô đun](#))

Trong RSA, việc sử dụng phép toán số học mô-đun để thực hiện mã hóa và giải mã. Trong ví dụ trên, ta có thể tính $n = p \times q = 3233 = 61 \times 53$. Đây là một số học mô-đun trong hệ thống mật mã RSA.

2.1.3 Phân tích thừa số nguyên tố

Thừa số nguyên tố là thừa số, nhưng là các số nguyên tố. Đây là quá trình phân tích một số thành tích của các số nguyên tố nhỏ hơn.

Trong RSA, việc phân tích thừa số nguyên tố là cơ sở để giải mã thông tin được mã hóa. Ví dụ, nếu chúng ta mã hóa một thông điệp bằng cách sử dụng khóa công khai e và nhận được một số mã hóa là $c = 3233$, ta có thể phân tích $c = 3233 = 61 \times 53$ thành tích của các số nguyên tố nhỏ hơn để giải mã.

2.1.4 Mật mã RSA

Mật mã RSA là một hệ thống mật mã công khai được phát minh bởi ba nhà khoa học là Ron Rivest, Adi Shamir và Leonard Adleman vào năm 1978. Tên của hệ thống này được đặt theo chữ cái đầu của các họ của ba nhà khoa học này. RSA là một trong những phương pháp mật mã công khai phổ biến nhất và được sử dụng rộng rãi trong các ứng dụng bảo mật, chẳng hạn trong mã hóa dữ liệu, giải mã dữ liệu, xác thực nguồn gốc của dữ liệu và tạo chữ ký số.

Mật mã RSA cung cấp tính bảo mật cao, vì việc giải mã dữ liệu mã hóa mà không có khóa bí mật tương ứng là rất khó đối với các kẻ tấn công. Nó cũng đảm bảo tính toàn vẹn của dữ liệu và xác thực nguồn gốc của dữ liệu thông qua việc sử dụng chữ ký số. Tuy nhiên, mật mã RSA cũng có nhược điểm của nó, chẳng hạn là tốc độ mã hóa và giải mã chậm đối với dữ liệu lớn và khó khăn trong việc quản lý các khóa công khai và khóa bí mật.

2.1.5 Nguyên tắc hoạt động

Mật mã RSA hoạt động dựa trên nguyên tắc của mã hóa và giải mã công khai, trong đó mỗi người dùng có một cặp khóa gồm khóa công khai và khóa bí mật. Khóa công khai được công khai cho các đối tượng khác trong hệ thống, trong khi khóa bí mật

chỉ được giữ bởi người dùng tạo ra nó. Quá trình mã hóa và giải mã trong RSA được thực hiện bằng phép toán mũ trong lý thuyết số và dựa trên tính khó của việc phân tích số nguyên tố lớn thành nhân tử.

Để rõ hơn, em chia cách thức hoạt động của hệ thống mật mã RSA gồm các bước:

- Bước 1: Tạo cặp khóa: Người dùng tạo ra một cặp khóa gồm khóa công khai và khóa bí mật. Khóa công khai được công khai cho các đối tượng khác trong hệ thống, trong khi khóa bí mật chỉ được giữ bí mật bởi người dùng tạo ra nó.
- Bước 2: Mã hóa dữ liệu: Người gửi dùng khóa công khai của người nhận để mã hóa dữ liệu cần gửi đi. Quá trình mã hóa này dựa trên phép toán mũ trên lý thuyết số, trong đó dữ liệu gốc được mũ bởi số nguyên n và sau đó lấy phần dư khi chia cho một số nguyên e làm khóa công khai.
- Bước 3: Gửi dữ liệu mã hóa: Dữ liệu sau khi mã hóa được gửi đi đến người nhận thông qua các kênh truyền thông công cộng.
- Bước 4: Giải mã dữ liệu: Người nhận dùng khóa bí mật của mình để giải mã dữ liệu nhận được. Quá trình giải mã này cũng dựa trên phép toán mũ trên lý thuyết số, trong đó dữ liệu mã hóa được mũ bởi số nguyên n và sau đó lấy phần dư khi chia cho một số nguyên d làm khóa bí mật.
- Bước 5: Nhận dữ liệu giải mã: Dữ liệu sau khi giải mã được người nhận nhận được và giải mã thành dữ liệu gốc.

2.1.6 Ưu điểm và nhược điểm

2.1.6.1 Ưu điểm

- Bảo mật cao: RSA dựa trên tính khó của việc phân tích số nguyên tố lớn thành nhân tử, là một bài toán tính toán phức tạp và đòi hỏi nhiều thời gian để thực hiện. Do đó, việc giải mã dữ liệu mã hóa mà không có khóa bí mật là rất khó đối với các kẻ tấn công.

- Khóa công khai có thể được công khai: Khóa công khai của RSA có thể được công khai cho mọi người trong hệ thống, giúp dễ dàng thiết lập kênh liên lạc bảo mật giữa các đối tượng trong hệ thống mà không cần trao đổi khóa trước.
- Sử dụng trong nhiều ứng dụng: RSA được sử dụng rộng rãi trong nhiều ứng dụng bảo mật, chẳng hạn như mã hóa thông tin trong giao tiếp qua mạng, chữ ký số, xác thực người dùng, và nhiều ứng dụng khác.

2.1.6.2 Nhược điểm

- Tốc độ mã hóa/giải mã chậm: Quá trình mã hóa và giải mã RSA có thể chậm đối với các dữ liệu lớn vì liên quan đến phép toán mũ trên số nguyên lớn, đòi hỏi khá nhiều tài nguyên tính toán.
- Kích thước khóa lớn: Để đạt được mức độ bảo mật cao, kích thước của khóa trong RSA cần phải lớn, đồng nghĩa với việc yêu cầu lưu trữ và xử lý khóa lớn, gây khó khăn trong việc triển khai trên các hệ thống có tài nguyên hạn chế.
- Nguy cơ tấn công trên khóa bí mật: Nếu khóa bí mật của RSA bị rò rỉ hoặc bị đánh cắp, thì tính bảo mật của hệ thống sẽ bị đe dọa, và dữ liệu có thể bị lộ.
- Khả năng tấn công bằng phương pháp tìm kiếm khóa: RSA có thể bị tấn công bằng cách tìm kiếm trực tiếp khóa bí mật dựa trên phép toán mũ, đòi hỏi đến phép toán tính toán độ phức tạp thấp hơn so với việc phân tích số nguyên tố, tuy nhiên vẫn là một khả năng đe dọa đối với tính bảo mật của RSA.

2.2 Thực hành

Em xây dựng thuật toán theo các bước đã nêu ở phần 2.1.5.

```

52100703.py x text.txt
52100703.py > ...
30 import math
31 #Bước 1: Chọn hai số nguyên tố lớn (ở bước này em sẽ chọn mặc định số 61 và 52)
32 p = 61
33 q = 53
34 #Bước 2: Tính n = p * q
35 n = p * q
36 #Bước 3: Chọn giá trị cho e (public key) (ở bước này em sẽ chọn mặc định số 17)
37 e = 17
38 #Step 4: Tính d bằng Thuật toán Euclide mở rộng (private key)
39 phi_n = (p-1) * (q-1)
40 d = ExtEuclidean(e, phi_n)

```


Hình 2.2.1 Thuật toán xây dựng hệ thống mật mã RSA ở bước 1, 2, 3, 4

```

41 #Bước 5: Xác định chức năng mã hóa và giải mã
42 #Chức năng mã hóa tin nhắn
43 def encrypt(message, n, e):
44     encrypted_msg = []
45     for char in message:
46         #Chuyển đổi ký tự thành giá trị ASCII
47         char_ascii = ord(char)
48         #Áp dụng công thức mã hóa RSA: C = M^e mod n
49         encrypted_char = pow(char_ascii, e, n)
50         encrypted_msg.append(encrypted_char)
51     return encrypted_msg
52 #Chức năng giải mã tin nhắn
53 def decrypt(encrypted_msg, n, d):
54     decrypted_msg = ""
55     for char_ascii in encrypted_msg:
56         #Áp dụng công thức giải mã RSA: M = C^d mod n
57         decrypted_char = pow(char_ascii, d, n)
58         #Chuyển đổi giá trị ASCII thành ký tự
59         decrypted_char = chr(decrypted_char)
60         decrypted_msg += decrypted_char
61     return decrypted_msg
62 #Bước 6: Mã hóa và Giải mã tin nhắn
63 #Tin nhắn được mã hóa
64 message = (input("Nhập tin nhắn cần mã hóa: "))
65 #Mã hóa tin nhắn
66 encrypted_message = encrypt(message, n, e)
67 print("Mã hóa tin nhắn:", encrypted_message)
68 #Giải mã tin nhắn
69 decrypted_message = decrypt(encrypted_message, n, d)
70 print("Giải mã tin nhắn:", decrypted_message)

```

Hình 2.2.2 Thuật toán xây dựng hệ thống mật mã RSA ở bước 5, 6

2.3 Kết quả đạt được

Với dữ liệu nhập vào từ bàn phím “My full name is Dinh Phuong My. I come from Bien Hoa City, Dong Nai. I am currently a 2nd year student of the Faculty of Information Technology, majoring in Computer Networking, Ton Duc Thang University. I'm doing a report for my subject, which is discrete structure.” thì kết quả mã hóa tin nhắn là [3123, 487, 1992, 1369, 2160, 745, 745, 1992, 2235, 1632, 2271, 1313, 1992, 3179, 1230, 1992, 1759, 3179, 2235, 2170, 1992, 2933, 2170, 2160, 2185, 2235, 2923, 1992, 3123, 487, 2825, 1992, 1486, 1992, 281, 2185, 2271, 1313, 1992, 1369, 2412, 2185, 2271, 1992, 524, 3179, 1313, 2235, 1992, 3000, 2185, 1632, 1992, 641, 3179, 884, 487, 678, 1992, 1759, 2185, 2235, 2923, 1992, 3165, 1632, 3179, 2825, 1992, 1486, 1992, 1632, 2271, 1992, 281, 2160, 2412, 2412, 1313, 2235, 884, 745, 487, 1992, 1632, 1992, 538, 2235, 1773, 1992, 487, 1313, 1632, 2412, 1992, 1230, 884, 2160, 1773, 1313, 2235, 884, 1992, 2185, 1369, 1992, 884, 2170, 1313, 1992, 325, 1632, 281, 2160,

745, 884, 487, 1992, 2185, 1369, 1992, 1486, 2235, 1369, 2185, 2412, 2271, 1632, 884, 3179, 2185, 2235, 1992, 2159, 1313, 281, 2170, 2235, 2185, 745, 2185, 2923, 487, 678, 1992, 2271, 1632, 1696, 2185, 2412, 3179, 2235, 2923, 1992, 3179, 2235, 1992, 641, 2185, 2271, 612, 2160, 884, 1313, 2412, 1992, 3165, 1313, 884, 1107, 2185, 2412, 690, 3179, 2235, 2923, 678, 1992, 2159, 2185, 2235, 1992, 1759, 2160, 281, 1992, 2159, 2170, 1632, 2235, 2923, 1992, 2310, 2235, 3179, 2578, 1313, 2412, 1230, 3179, 884, 487, 2825, 1992, 1486, 1956, 2271, 1992, 1773, 2185, 3179, 2235, 2923, 1992, 1632, 1992, 2412, 1313, 612, 2185, 2412, 884, 1992, 1369, 2185, 2412, 1992, 2271, 487, 1992, 1230, 2160, 2570, 1696, 1313, 281, 884, 678, 1992, 1107, 2170, 3179, 281, 2170, 1992, 3179, 1230, 1992, 1773, 3179, 1230, 281, 2412, 1313, 884, 1313, 1992, 1230, 884, 2412, 2160, 281, 884, 2160, 2412, 1313] và kết quả giải mã tin nhắn: My full name is Dinh Phuong My. I come from Bien Hoa City, Dong Nai. I am currently a 2nd year student of the Faculty of Information Technology, majoring in Computer Networking, Ton Duc Thang University. I'm doing a report for my subject, which is discrete structure.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
```

```
PS C:\WINDOWS\System32\WindowsPowerShell\v1.0> & "C:/Program Files/Python/python.exe" "h:/My Drive/The Second Year/[HK2] 501044 Cấu trúc rời rạc/Thực hành/Assesment/52100783.py"
Nhập tin nhân cần mã hóa: My full name is Dinh Phuong My. I come from Bien Hoa City, Dong Nai. I am currently a 2nd year student of the Faculty of Information Technology, majoring in Computer Networking, Ton Duc Thang University. I'm doing a report for my subject, which is discrete structure
Mã hóa tin nhắn:
2933, 2170, 2160, 2185, 2235, 2923, 1992, 3123, 487, 2825, 1992, 1486, 1992, 281, 2185, 2271, 1313, 1992, 1369, 2412, 2185, 2271, 1992, 524
3, 3179, 1313, 2235, 1992, 3000, 2185, 1632, 1992, 641, 3179, 884, 487, 678, 1992, 1759, 2185, 2235, 2923, 1992, 3165, 1632, 3179, 2825, 1992
1486, 1992, 1632, 2271, 1992, 281, 2160, 2412, 2412, 1313, 2235, 884, 745, 487, 1992, 1632, 1992, 538, 2235, 1773, 1992, 487, 1313, 1632,
2412, 1992, 1230, 884, 2160, 1773, 1313, 2235, 884, 1992, 2185, 1369, 1992, 2271, 884, 2170, 1313, 1992, 325, 1632, 281, 2160, 745, 884, 487, 1992
2185, 1369, 1992, 1486, 2235, 1369, 2185, 2412, 2271, 1632, 884, 3179, 2185, 2235, 1992, 2159, 1313, 281, 2170, 2235, 2185, 745, 2185, 292
3, 487, 678, 1992, 2271, 1632, 1696, 2185, 2412, 3179, 2235, 2923, 1992, 3179, 2235, 1992, 641, 2185, 2271, 612, 2160, 884, 1313, 2412, 1992
3165, 1313, 884, 1107, 2185, 2412, 690, 3179, 2235, 2923, 678, 1992, 2159, 2185, 2235, 1992, 1759, 2160, 281, 1992, 2159, 2170, 1632, 2235
2923, 1992, 2310, 2235, 3179, 2578, 1313, 2412, 1230, 3179, 884, 487, 2825, 1992, 1486, 1992, 2271, 1992, 1773, 2185, 3179, 2235, 2923, 19
92, 1632, 1992, 2412, 1313, 612, 2185, 2412, 884, 1992, 1369, 2185, 2412, 1992, 2271, 487, 1992, 1230, 2160, 2570, 1696, 1313, 281, 884, 678
1992, 1107, 2170, 3179, 281, 2170, 1992, 3179, 1230, 1992, 1773, 3179, 1230, 281, 2412, 1313, 884, 1313, 1992, 1230, 884, 2412, 2160, 281,
884, 2160, 2412, 1313]
Giải mã tin nhắn: My full name is Dinh Phuong My. I come from Bien Hoa City, Dong Nai. I am currently a 2nd year student of the Faculty of I
nformation Technology, majoring in Computer Networking, Ton Duc Thang University. I'm doing a report for my subject, which is discrete struc
ture
PS C:\WINDOWS\System32\WindowsPowerShell\v1.0>
```

Hình 2.3 Kết quả mã hóa và giải mã bằng hệ thống mật mã RSA thành công

2.4 Phân tích hiệu quả và tính bảo mật

2.4.1 Hiệu quả

Mã hóa và giải mã trong hệ thống RSA có tính đối xứng, tức là mỗi phép mã hóa cần một phép giải mã tương ứng và ngược lại. Điều này làm cho RSA khá chậm trong

việc mã hóa và giải mã dữ liệu lớn vì các phép toán mũ và phép chia lấy dư có độ phức tạp tính toán cao.

Tuy nhiên, một ưu điểm của RSA là quá trình tạo khóa và phân phối khóa công khai rất đơn giản và nhanh chóng. Khóa công khai có thể được chia sẻ công khai mà không cần sự đồng thuận trước của người nhận tin nhắn. Điều này đồng nghĩa với việc không cần một kênh bảo mật riêng biệt để chuyển giao khóa.

2.4.2 Tính bảo mật

Được xây dựng trên cơ sở tính chất toán học của tích hai số nguyên tố lớn, việc giải mã tin nhắn mã hóa RSA dựa trên khó khăn của bài toán phân tích số nguyên tố lớn thành nhân tử, mà hiện tại vẫn chưa có thuật toán hiệu quả để giải quyết trong thời gian hợp lý.

Độ dài của khóa trong RSA đóng vai trò quan trọng trong độ an toàn của hệ thống. Với khóa có độ dài đủ lớn, hệ thống RSA có thể cung cấp mức độ bảo mật rất cao. Tuy nhiên, với sự phát triển của tính toán và công nghệ, các cuộc tấn công dựa trên lý thuyết số nguyên tố hoặc lý thuyết đồng dư có thể trở nên hiệu quả hơn trong tương lai, do đó việc chọn độ dài khóa là một yếu tố quan trọng trong tính bảo mật của hệ thống RSA.

Một số các cuộc tấn công khác như tấn công theo phương pháp tấn công bội số chung, tấn công khoảng cách đệm (timing attack), hay tấn công bằng cách sử dụng đồng dư, v.v. cũng có thể đe dọa tính bảo mật của hệ thống RSA nếu không được triển khai đúng cách và cập nhật thường xuyên để đối phó với các mối đe dọa này.

Ngoài ra, hiệu quả và tính bảo mật của hệ thống RSA còn phụ thuộc vào độ dài của tin nhắn cần được mã hóa. Nếu tin nhắn quá dài, việc mã hóa RSA có thể trở nên chậm chạp do phải thực hiện nhiều phép tính mũ. Một số giải pháp đã được đưa ra để cải thiện tốc độ mã hóa và giải mã của RSA, chẳng hạn như RSA đệ quy (RSA-CRT) và RSA song song. Tuy nhiên, những giải pháp này cần được triển khai cẩn thận để tránh gây ra lỗ hổng bảo mật.

Tóm lại, hệ thống mật mã RSA đã triển khai là một giải pháp mã hóa công khai phổ biến và có tính bảo mật cao, tuy nhiên nó cũng có nhược điểm về hiệu quả tính toán. Để đảm bảo tính bảo mật và hiệu quả của hệ thống RSA, cần chọn độ dài khóa phù hợp, triển khai đúng cách, và định kỳ cập nhật để đối phó với các mối đe dọa mới nhất trong lĩnh vực bảo mật thông tin.

2.5 Kết luận

Mặc dù hệ thống mật mã RSA là một trong những hệ thống mật mã công khai phổ biến và có tính bảo mật cao, nó cũng đối diện với một số mối đe dọa bảo mật tiềm ẩn và hạn chế, bao gồm các công nghệ tính toán phát triển, độ dài khóa, tấn công bằng cách sử dụng đặc điểm cấu trúc và quá trình phân phối khóa công khai.

Để cải thiện việc triển khai hệ thống mật mã RSA, cần phải sử dụng độ dài khóa dài hơn để tăng cường bảo mật. Khi sức mạnh tính toán tăng lên, các khóa dài hơn là cần thiết để chống lại các cuộc tấn công vũ phu. Độ dài khóa ít nhất là 2048 bit được khuyến nghị cho hầu hết các ứng dụng, trong khi 3072 bit trở lên được khuyến nghị cho dữ liệu nhạy cảm. Sử dụng nguồn ngẫu nhiên thực sự ngẫu nhiên và không thể đoán trước để tạo khóa và giá trị ngẫu nhiên trong thuật toán RSA. Tránh sử dụng các trình tạo số ngẫu nhiên yếu hoặc có thể dự đoán được vì chúng có thể ảnh hưởng đến tính bảo mật của mã hóa RSA. Thực hiện các biện pháp quản lý khóa an toàn, bao gồm lưu trữ an toàn, sao lưu và chuyển giao khóa riêng. Giới hạn số lượng cá nhân có quyền truy cập vào khóa riêng và thường xuyên cập nhật cũng như xoay vòng khóa khi cần. Bảo vệ chống lại các cuộc tấn công kênh phụ, chẳng hạn như tấn công thời gian, tấn công phân tích năng lượng và tấn công điện từ, nhằm khai thác thông tin bị rò rỉ trong các hoạt động mã hóa. Sử dụng các biện pháp đối phó, chẳng hạn như triển khai liên tục và mô-đun bảo mật phần cứng (HSM), để bảo vệ chống lại các cuộc tấn công này.

Tóm lại, hệ thống mật mã RSA là một công nghệ mã hóa công khai phổ biến và có tính bảo mật cao, cần cải thiện những điều trên để hệ thống mật mã RSA không bị dính vào mối đe dọa bảo mật và hạn chế.

TÀI LIỆU THAM KHẢO

Tiếng Việt

1. Thái Thanh Tùng, *Giáo trình An ninh mạng và bảo mật dữ liệu*, Đại học Mở Hà Nội, 2006.
2. Wikipedia tiếng Việt, *Số học Mô đun*, truy cập vào ngày 10 tháng 4 năm 2023.
3. Wikipedia tiếng Việt, *RSA (mã hóa)*, truy cập vào ngày 10 tháng 4 năm 2023.

Tiếng Anh

4. 101 Computing, *Symmetric vs. Asymmetric Encryption*, 101 Computing, truy cập vào ngày 10 tháng 4 năm 2023.
5. 101 Computing, *Euclid's Division Algorithm*, 101 Computing, truy cập vào ngày 10 tháng 4 năm 2023.