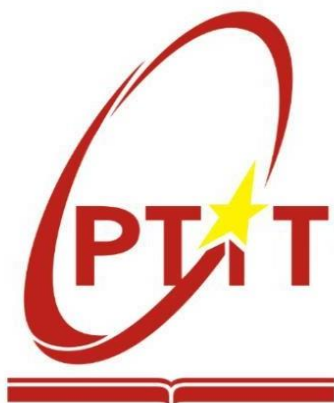


**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  
**KHOA AN TOÀN THÔNG TIN**

---



**BÁO CÁO THỰC HÀNH**

**Bài 16: Lập trình thuật toán mật mã học**

**Họ và tên: VũThành Long**

**Mã sinh viên: B21DCAT012**

**Nhóm: 06**

**Môn học: Thực tập cơ sở**

**Giảng viên giảng dạy: Nguyễn Hoa Cương**

**Hà Nội, 2024**

I.Mục đích.....	2
II.Tìm hiểu lý thuyết.....	2
1. Lập trình với số lớn.....	2
2. Giải thuật mật mã khóa công khai RSA .....	2
III.Các bước thực hiện và kết quả .....	3
IV.Tài liệu tham khảo :.....	6

## Bài 16: Lập trình thuật toán mật mã học

### I. Mục đích

Sinh viên tìm hiểu một giải thuật mã hóa phổ biến và lập trình được chương trình mã hóa và giải mã sử dụng ngôn ngữ lập trình phổ biến như C/C++/Python/Java, đáp ứng chạy được với số lớn.

### II. Tìm hiểu lý thuyết

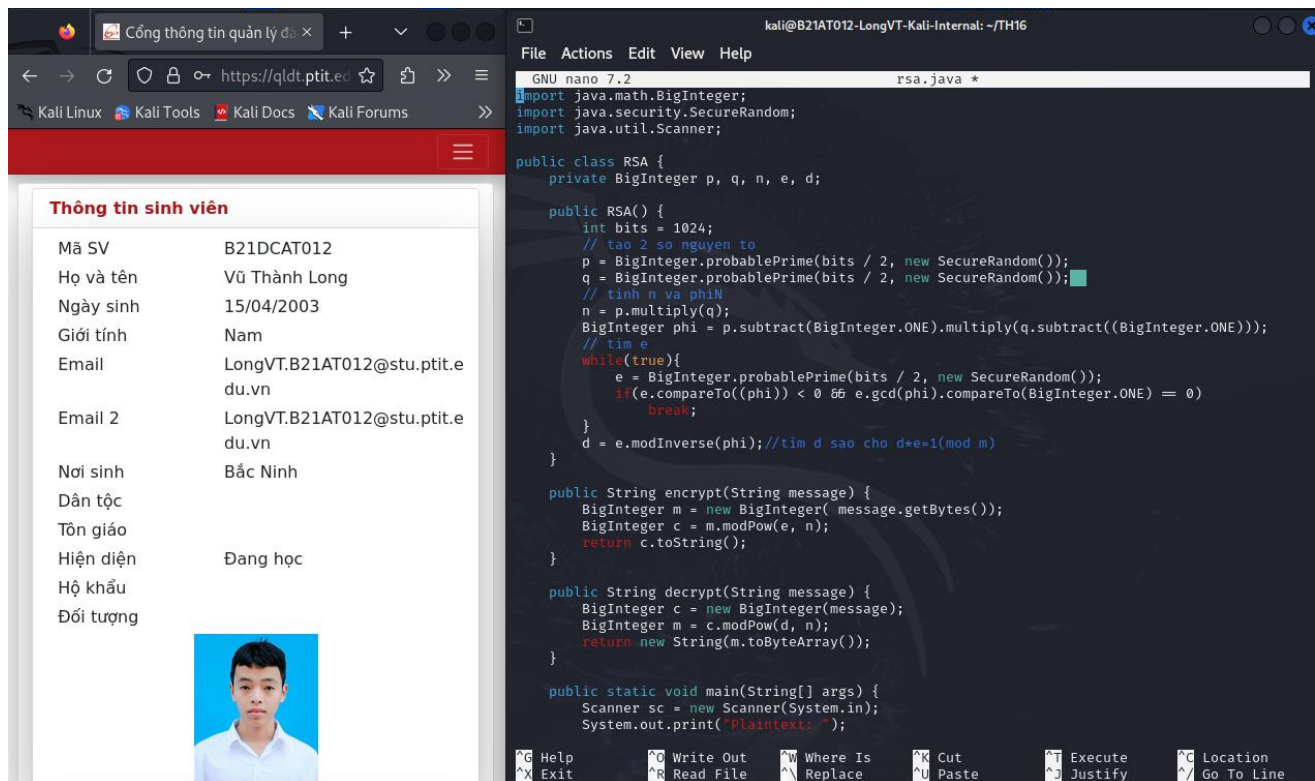
#### 1. Lập trình với số lớn

- Việc lập trình với những số có độ dài hàng nghìn bit là rất khó khăn. Thay vì tự viết hàm tính toán các số lớn, sinh viên sử dụng class BigInteger có sẵn trong Java chuyên để xử lý các số lớn.
- Class BigInteger cũng cung cấp những phép toán cơ bản như cộng add(), trừ subtract(), nhân multiply(), chia divide(), giúp việc tính toán các phép toán cơ bản dễ dàng hơn.
- Ngoài ra, BigInteger còn cung cấp hàm lũy thừa lấy phần dư modpow() hay hàm nghịch đảo modulo modInverse() giúp việc lập trình mã hoá và giải mã RSA dễ dàng hơn.

#### 2. Giải thuật mật mã khóa công khai RSA

- Thuật toán mã hoá RSA là thuật toán mã hoá khóa công khai được sử dụng rộng rãi để truyền dữ liệu an toàn.
- Thuật toán mã hoá RSA được phát triển bởi Rivest, Shamir, Adleman.  
Quy trình mã hoá của RSA được công khai năm 1977.
- Độ an toàn của RSA liên hệ chặt chẽ với độ khó của bài toán phân tích nhân tử của một số rất lớn thành hai thừa số nguyên tố. Hiện nay vẫn chưa có siêu máy tính nào có thể giải bài toán này với thời gian chấp nhận được, nhưng trong tương lai với máy tính lượng tử có thể sẽ khả thi.
- Quy trình mã hoá:
  - + Chọn hai số nguyên tố lớn  $p$  và  $q$  và tính  $N = pq$ . Cần chọn  $p$  và  $q$  sao cho  $M < 2^{(i-1)} < N < 2^i$
  - + Tính  $\Phi(n) = (p - 1)(q - 1)$
  - + Tìm một số  $e$  sao cho:  $\{e \text{ và } \Phi(n) \text{ là 2 số cùng nhau và } 0 < e < \Phi(n)\}$
  - + Tìm một số  $d$  sao cho:  $e \cdot d \equiv 1 \pmod{\Phi(n)}$  (hay:  $d = e^{-1} \pmod{\Phi(n)}$ )
  - + Chọn khóa công khai  $K1$  là cặp  $(e, N)$ , khóa riêng  $K2$  là cặp  $(d, N)$ .
  - + Mã hoá  $C = M^e \pmod{N}$ , hoặc  $C = M^d \pmod{N}$  nếu mã hoá chứng thực.
  - + Giải mã  $M = C^d \pmod{N}$ , hoặc  $M = C^e \pmod{N}$  nếu chứng thực.

### III. Các bước thực hiện và kết quả



The screenshot displays a Kali Linux desktop environment. On the left, a web browser window shows a student profile page titled "Thông tin sinh viên" (Student Information). The profile details are as follows:

Thông tin sinh viên	
Mã SV	B21DCAT012
Họ và tên	Vũ Thành Long
Ngày sinh	15/04/2003
Giới tính	Nam
Email	LongVT.B21AT012@stu.ptit.edu.vn
Email 2	LongVT.B21AT012@stu.ptit.edu.vn
Nơi sinh	Bắc Ninh
Dân tộc	
Tôn giáo	
Hiện diện	Đang học
Hộ khẩu	
Đối tượng	

Below the text is a small portrait photo of a young man with short black hair, wearing a light blue shirt, against a blue background.

On the right, a terminal window titled "kali@B21AT012-LongVT-Kali-Internal: ~/TH16" shows the GNU nano 7.2 editor editing a file named "rsa.java". The code is a Java implementation of the RSA algorithm, including key generation, encryption, and decryption methods. The code is as follows:

```
GNU nano 7.2 rsa.java *
import java.math.BigInteger;
import java.security.SecureRandom;
import java.util.Scanner;

public class RSA {
    private BigInteger p, q, n, e, d;

    public RSA() {
        int bits = 1024;
        // tạo 2 số nguyên tố
        p = BigInteger.probablePrime(bits / 2, new SecureRandom());
        q = BigInteger.probablePrime(bits / 2, new SecureRandom());
        // tính n và phiN
        n = p.multiply(q);
        BigInteger phi = p.subtract(BigInteger.ONE).multiply(q.subtract(BigInteger.ONE));
        // tìm e
        while(true){
            e = BigInteger.probablePrime(bits / 2, new SecureRandom());
            if(e.compareTo(phi) < 0 && e.gcd(phi).compareTo(BigInteger.ONE) == 0)
                break;
        }
        d = e.modInverse(phi); // tìm d sao cho d*e=1(mod n)
    }

    public String encrypt(String message) {
        BigInteger m = new BigInteger(message.getBytes());
        BigInteger c = m.modPow(e, n);
        return c.toString();
    }

    public String decrypt(String message) {
        BigInteger c = new BigInteger(message);
        BigInteger m = c.modPow(d, n);
        return new String(m.toByteArray());
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Plaintext: ");
    }
}
```

The terminal window also shows a menu at the bottom with various shortcuts: ^G Help, ^X Exit, ^O Write Out, ^R Read File, ^W Where Is, ^N Replace, ^K Cut, ^U Paste, ^T Execute, ^J Justify, ^C Location, and ^\_ Go To Line.

- Sử dụng thư viện BigInteger trong java để triển khai mã hóa RSA:
- + p, q sẽ là 2 số nguyên tố ngẫu nhiên
- + Hàm modInverse(): tính nghịch đảo của e trong modulo  $\Phi(n)$
- Thuật toán mã hóa và giải mã: thông tin mã hóa cần phải được chuyển thành dạng byte vì các thông tin được truyền thường ở dạng chuỗi ký tự

The screenshot shows a Kali Linux desktop environment. On the left, a web browser displays a form titled "Thông tin sinh viên" (Student Information) with the following details:

Mã SV	B21DCAT012
Họ và tên	Vũ Thành Long
Ngày sinh	15/04/2003
Giới tính	Nam
Email	LongVT.B21AT012@stu.ptit.edu.vn
Email 2	LongVT.B21AT012@stu.ptit.edu.vn
Nơi sinh	Bắc Ninh

On the right, a terminal window shows the code for `rsa.java`:

```

GNU nano 7.2      rsa.java *
import java.math.BigInteger;
import java.security.SecureRandom;
import java.util.Scanner;

public class RSA {
    private BigInteger p, q, n, e, d;

    public RSA() {
        int bits = 1024;
        // tạo 2 số nguyên tố
        p = BigInteger.probablePrime(bits / 2, new SecureRandom());
        q = BigInteger.probablePrime(bits / 2, new SecureRandom());
        // tính n và phiN
        n = p.multiply(q);
        BigInteger phi = p.subtract(BigInteger.ONE).multiply(q.subtract(BigInteger.ONE));
        // tìm e
        while(true){
            e = BigInteger.probablePrime(bits / 2, new SecureRandom());
            if(e.compareTo(phi) < 0 && e.gcd(phi).compareTo(BigInteger.ONE) == 0)
                break;
        }
        d = e.modInverse(phi); // tìm d sao cho d*e=1(mod m)
    }
}

```

- Thử nghiệm với số lớn:

The screenshot shows the same Kali Linux desktop environment. The web browser on the left now displays a more complete form titled "Thông tin sinh viên" (Student Information) with the following details:

Mã SV	B21DCAT012
Họ và tên	Vũ Thành Long
Ngày sinh	15/04/2003
Giới tính	Nam
Email	LongVT.B21AT012@stu.ptit.edu.vn
Email 2	LongVT.B21AT012@stu.ptit.edu.vn
Nơi sinh	Bắc Ninh
Dân tộc	
Tôn giáo	
Hiện diện	Đang học
Hộ khẩu	
Đối tượng	

On the right, the terminal window shows the execution of `java rsa.java` and its output:

```

(kali@B21AT012-LongVT-Kali-Internal:~/TH16)
$ java rsa.java
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Plaintext: 566356463465431324214641523432141242112142
Encrypted message: 625824859475724989832582663819733659433902050216548490958355823190020468454.
1140692512222007490656926769589385179469864923143249796666873316961459063719469630251651392153.
1569334009944161903911422838178843863437023618506396374375839633486087526846831637659703689465.
684084101151198233180028452565361420826
Decrypted message: 566356463465431324214641523432141242112142

```

## - Thử nghiệm mã hóa và giải mã chuỗi ký tự: “I am B21DCAT012”

The screenshot displays a Kali Linux desktop environment. On the left, a web browser window is open to a page titled "Thông tin sinh viên" (Student Information). The page contains a form with the following details:

Thông tin sinh viên	
Mã SV	B21DCAT012
Họ và tên	Vũ Thành Long
Ngày sinh	15/04/2003
Giới tính	Nam
Email	LongVT.B21AT012@stu.ptit.edu.vn
Email 2	LongVT.B21AT012@stu.ptit.edu.vn
Nơi sinh	Bắc Ninh
Dân tộc	
Tôn giáo	
Hiện diện	Đang học
Hộ khẩu	
Đối tượng	

Below the form is a small portrait photo of a young man with short black hair, wearing a light blue shirt, against a blue background.

On the right, a terminal window is open, showing the execution of a Java program. The terminal output is as follows:

```
kali@B21AT012-LongVT-Kali-Internal: ~/TH16
$ java rsa.java
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Plaintext: I am B21DCAT012
Encrypted message: 1075120619694393469156971007963270096875960886626997505028044819031227751058
308341480157201096530278354775335788069644653263065359847496903006901382621486852774420453965284
255066989893124343853387505140194101830165606394639278391456432229707288935449018063959777020360
7365944986028176234977349209766151537586
Decrypted message: I am B21DCAT012
```

#### **IV. Tài liệu tham khảo :**

- Triển khai thuật toán RSA bằng java : [https://youtu.be/fDhBBu\\_y7L4](https://youtu.be/fDhBBu_y7L4)
- Bài giảng Mật mã học cơ sở, thầy Đỗ Xuân Chợt.