BÁO CÁO THỰC HÀNH

**Bài thực hành số 02**

**MÃ HÓA KHOÁ CÔNG HAI**

|  |  |
| --- | --- |
| **Môn học:** An toàn Mạng máy tính | **Điểm tự đánh giá** |
| **Lớp:** NT101.O12.MMCL.1  **Giảng viên hướng dẫn:** Văn Thiên Luân | **10** |

**PHÂN CÔNG NHIỆM VỤ NHÓM 4:**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và Tên** | **Nhiệm vụ** |
| 20521633 | Nguyễn Tấn Phương Nam | * Được phần bài 1 phần 3 và 5 và viết báo cáo phần đó. * Hỗ trợ nhóm làm phần giao diện Form main của cả bài 1. |
|  |  |  |
|  |  |  |

**ĐÁNH GIÁ KHÁC:**

|  |  |
| --- | --- |
| Tổng thời gian thực hiện | 72 tiếng |
| Ý kiến *(nếu có)*  + Khó khăn  + Đề xuất, kiến nghị |  |

**MỤC LỤC**

[I. BÁO CÁO CHI TIẾT 3](#_Toc118326085)

[**1.** **Lý thuyết số học** 3](#_Toc118326086)

[*a.* *Mô tả hàm* 3](#_Toc118326087)

[*b.* *Chạy thử và đánh giá* 9](#_Toc118326088)

[**2.** **Mã hóa khóa công khai RSA** 12](#_Toc118326089)

[*a.* *Mô tả thuật toán* 12](#_Toc118326090)

[*b.* *Kiểm thử chương trình* 12](#_Toc118326091)

[**3.** **Ứng dụng chat có bảo mật** 14](#_Toc118326092)

[*a.* *Mô tả hoạt động của ứng dụng chat có mã hóa bằng RSA* 14](#_Toc118326093)

[*b.* *Một số đoạn code quan trọng* 16](#_Toc118326094)

[II. TÀI LIỆU THAM KHẢO 19](#_Toc118326095)

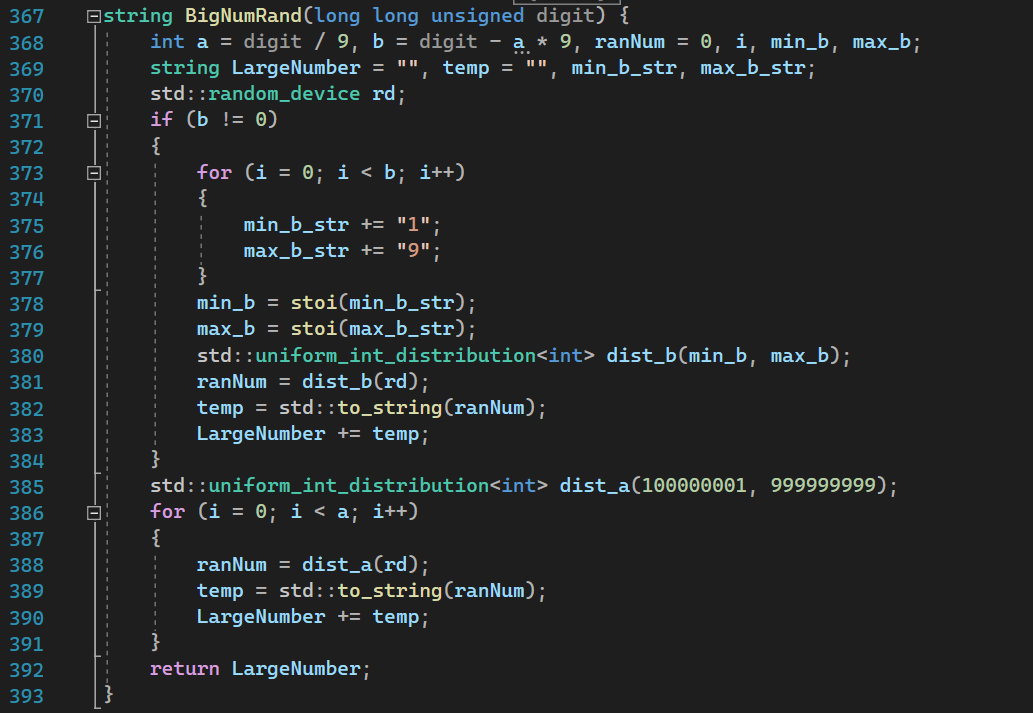
1. **BÁO CÁO CHI TIẾT**
2. **Lý thuyết số học**

Ngôn ngữ: C++

Thư viện ngôn ngữ:

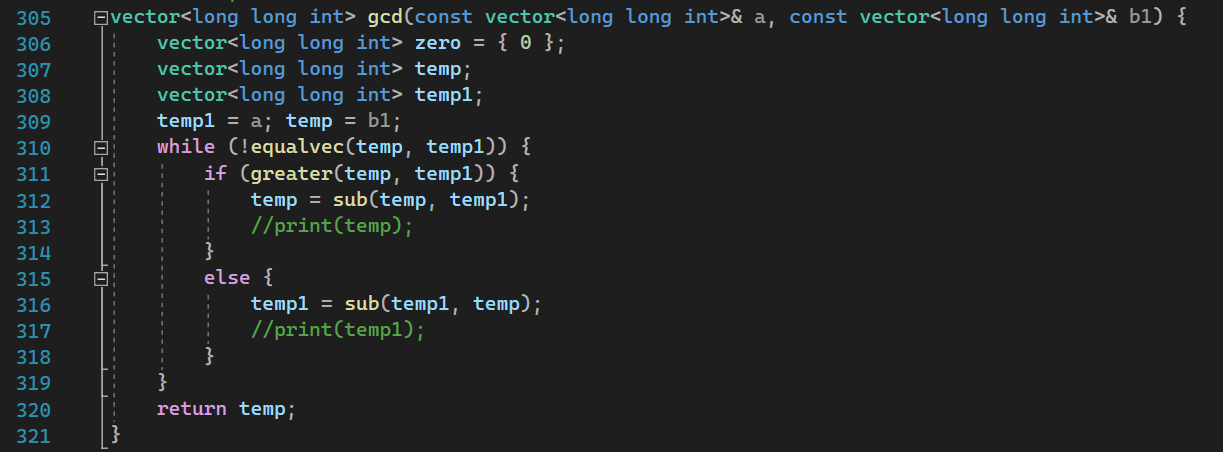
* string: tương tác với chuỗi
* math.h: sử dụng các phép tính số học
* ctime: theo dõi thời gian thực thi
* vector: tương tác với số lớn
* random: tạo ngẫu nhiên các số nhỏ

1. *Mô tả hàm*
2. Tạo ngẫu nhiên một số nguyên lớn

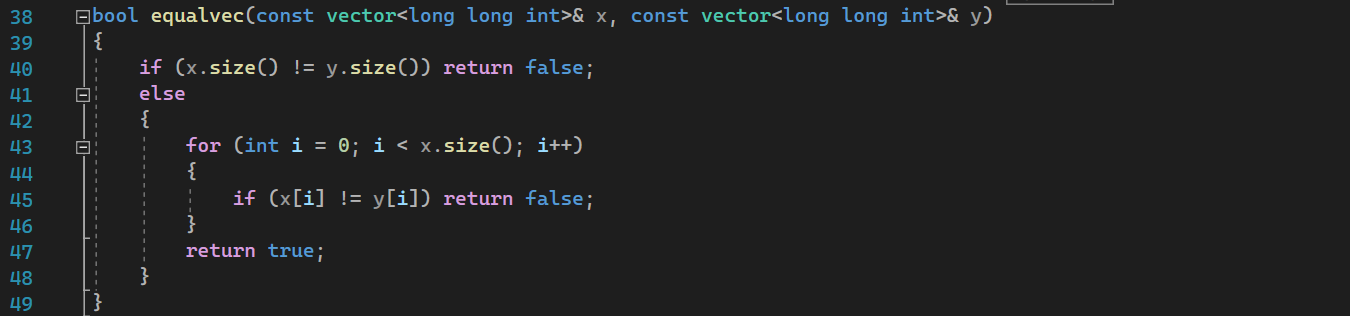


* Tạo ngẫu nhiên một dãy số có 9 chữ số từ 100000001 đến 999999999
* Ghép các dãy số lại với nhau cho đến khi được độ dài đã truyền vào trước đó
* Trả về dãy số đã được ghép, ta sẽ được một số nguyên lớn

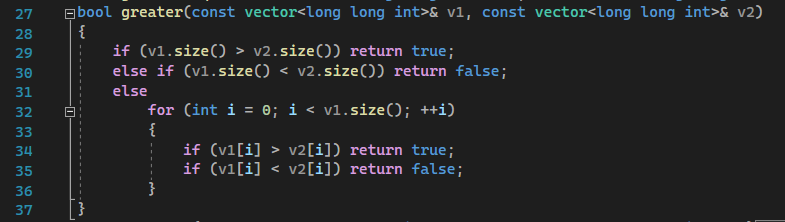
1. Tính GCD của 2 số nguyên lớn

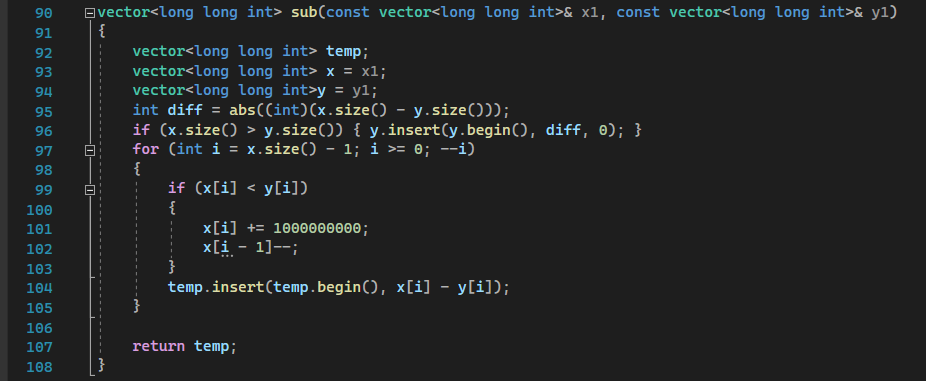


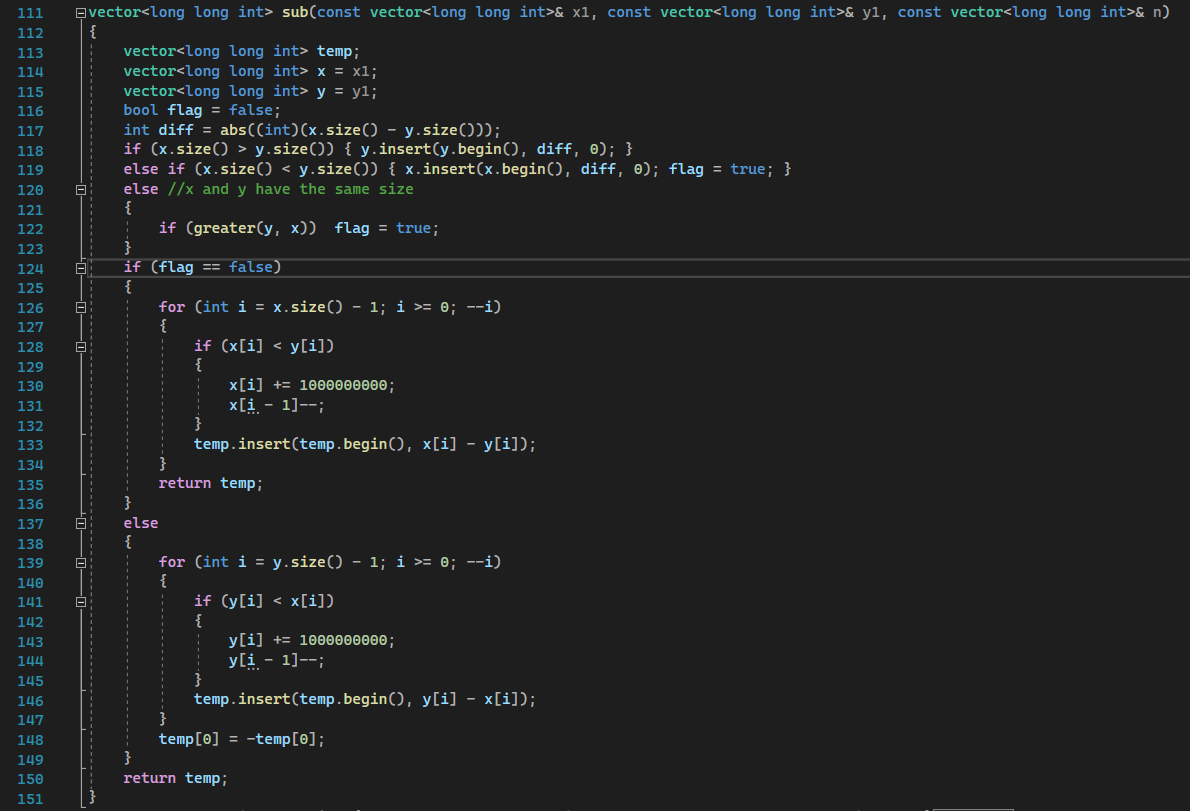
* Truyền vài 2 số nguyên lớn bằng cách tạo ngẫu nhiên từ hàm trên
* Tạo 2 biến khác và gán giá trị đã truyền vào ở trên vào 2 biến vừa tạo
* So sánh 2 số có khác nhau không bằng hàm **equalvec** và thêm **!** trước hàm đó



* So sánh 2 số với nhau bằng hàm **greater**, số nào lớn hơn sẽ lấy số đó trừ cho số còn lại (dùng hàm **sub**) cho đến khi nào chúng bằng nhau thì trả về giá trị đó.







1. Tính lũy thừa môđun ax *mod* p. Chương trình của bạn có thể tính toán trong trường hợp số mũ “lớn” (x > 80), ví dụ:794 *mod* 19.

A screenshot of a computer

Description automatically generated

**Hình 1.3.1: Phần giao diện tính lũy thừa modun ax mod p và đo thời gian trung bình của chương trình**

* Khi vào Form Main người dùng sẽ đứa chuột vào “Mở” và chọn bài 3 để mở chương trình.

A white screen with a black border

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated

* Người dùng sẽ nhập giá trị vào từng ô, nếu người dùng nhập không phải nguyên hoặc không nhập thì sẽ thông báo lỗi và yêu cầu người dùng nhập lại.
* Sau đó, người dùng sẽ nhấn button “Tính” để hiển thị kết quả ra màn hình. Người dùng sau nhiều lần thử với nhiều giá trị thì bấm vào button ”Tính Average” để hiển thị thời gian trung bình của chương trình.

A screenshot of a computer

Description automatically generated

**Hình 1.3.2: Đoạn mã sử dụng để tính lũy thừa modun ax mod p**

* Đoạn code được sử dụng để tính giá trị ax mod p cho trường hợp x lớn. Cách hoạt động của đoạn code trên:

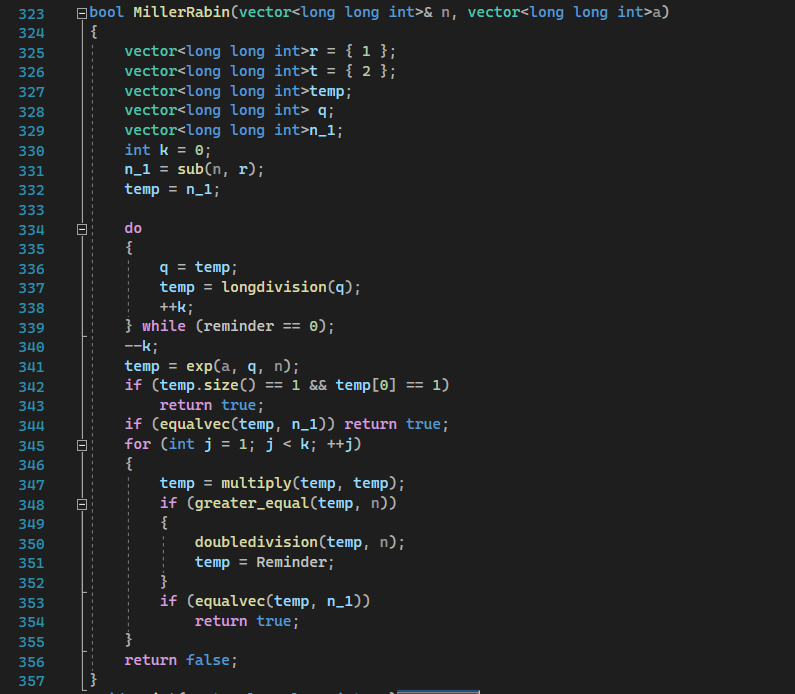
1. Kiểm tra xem p có bằng1: Nếu p bằng 1, thì kết quả luôn là 0 bởi vì bất kỳ số nào mod 1 đều bằng 0. Do đó, nếu p == 1, nó sẽ trả về 0.
2. Khởi tạo biến result = 1: result là biến sẽ chứa kết quả của phép lũy thừa mod.
3. Trước khi bắt đầu lũy thừa, a được giảm xuống bằng cách thực hiện phép toán a = a % p. Điều này đảm bảo rằng a luôn nằm trong khoảng từ 0 đến p-1.
4. Tiếp theo lặp while được sử dụng để thực hiện phép lũy thừa. Mỗi lần lặp, nó kiểm tra xem bit cuối cùng của x có phải là 1 không (x % 2 == 1). Nếu là 1, nó nhân result với a và sau đó lấy phần dư khi chia cho p. Sau đó, a được nhân với chính nó và lấy phần dư khi chia cho p. Cuối cùng, x được chia cho 2 để lấy bit tiếp theo. Điều này tiếp tục cho đến khi x trở thành 0.
5. Sau khi vòng lặp kết thúc, result chứa giá trị ax mod p và trả về giá trị result.

A screen shot of a computer program

Description automatically generated

**Hình 1.3.3: Đoạn mã sử dụng để bắt sự kiện click tính lũy thừa modun ax mod, thời gian trung bình của chương trình và hiển thị nó lên giao diện.**

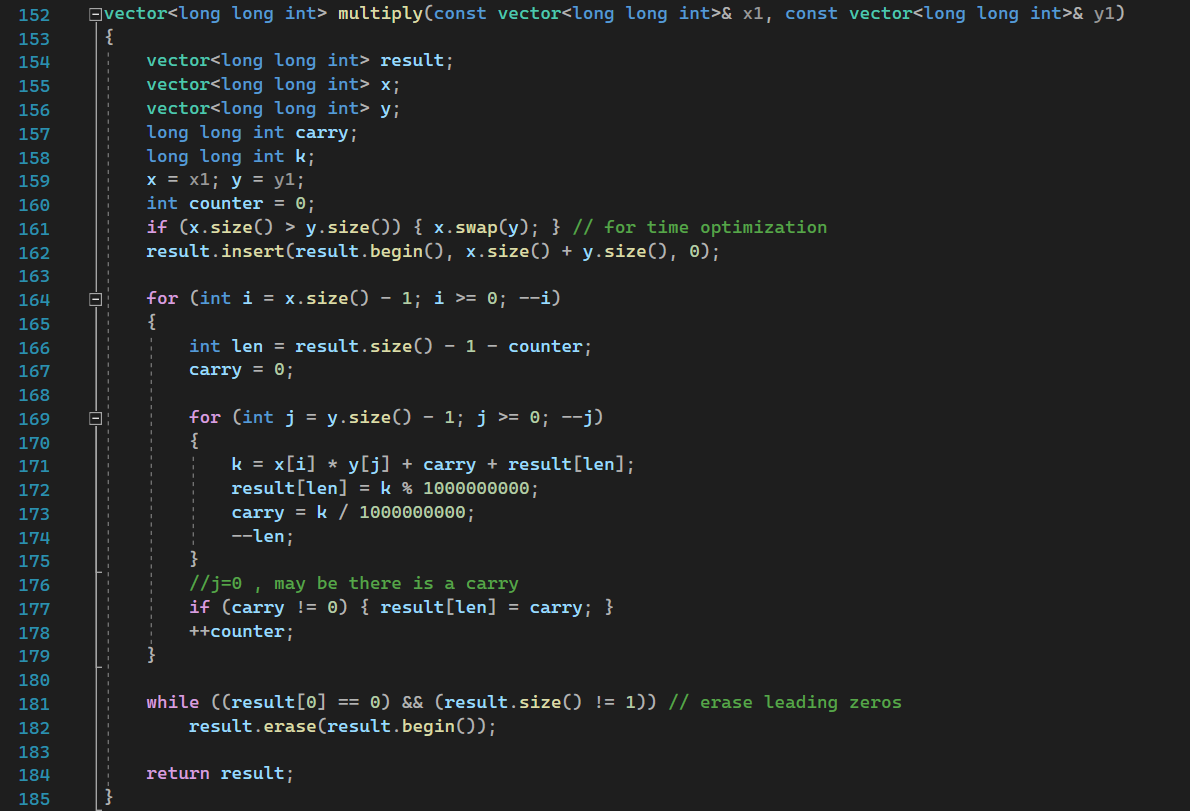
1. Thuật toán MillerRabin



* Phân tích n−1 thành dạng 2x×d
* Thử lần lượt các số a trong bảng sau:

|  |  |
| --- | --- |
| Giới hạn của n | Các số a cần dùng |
| n < 2,047 | 2 |
| n < 1,373,653 | 2, 3 |
| n < 9,080,191 | 31, 73 |
| n < 4,759,123,141 | 2, 7, 61 |
| n < 1,122,004,669,633 | 2, 13, 23, 1662803 |
| n < 2,152,302,898,747 | 2, 3, 5, 7, 11 |
| n < 3,474,749,660,383 | 2, 3, 5, 7, 11, 13 |
| n < 341,550,071,728,321 | 2, 3, 5, 7, 11, 13, 17 |
| n < 3,825,123,056,546,413,051 | 2, 3, 5, 7, 11, 13, 17, 19, 23 |
| n < 318,665,857,834,031,151,167,461 | 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37 |
| n < 3,317,044,064,679,887,385,961,981 | 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41 |

* Sau đó tiếp tục kiểm tra xem số truyền vào có phải số nguyên tố không bằng các hàm **equalvec, multiply, greater\_equal, doubledividision**.



A screenshot of a computer

Description automatically generated with medium confidence

1. *Chạy thử và đánh giá*
2. Tạo ngẫu nhiên một số nguyên lớn

* Có thể chạy không giới hạn, nhưng đối với chuỗi số gồm 500 chữ số thì thời giant trung bình ẽ vào khoảng 0.002 giây.

A screenshot of a computer

Description automatically generated with medium confidence

Text

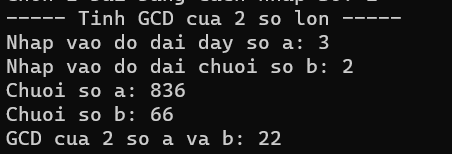
Description automatically generated

Text

Description automatically generated

1. Tính GCD của 2 số lớn

* Ta sẽ kiểm tra tính đúng đắn với số nhỏ trước



Text

Description automatically generated

* Đối với các số có cùng độ dài thì sẽ tốn tầm 0.15 giây

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

* Đối với các số có sự chênh lệch lớn hơn sẽ tốn nhiều thời gian hơn, tùy vào độ chênh lệch của các số.

1. Tính ax mod p

* Kiểm tra với ví dụ ban đầu và thử lại với các giá trị khác nhau:

A screenshot of a computer

Description automatically generated

**Hình b.1.3.1: Hình giao diện với thử lần 1 với ví dụ và tốn thời gian thực thi là 0.2903 ms, bộ nhớ sử dụng là 40075264 bytes, CPU time là 3281250 ticks.**

A screenshot of a computer

Description automatically generated

**Hình b.1.3.2: Hình giao diện với thử lần 2 và tốn thời gian thực thi là 0.0111 ms, bộ nhớ sử dụng là 40181760 bytes, CPU time là 5781250 ticks và tổng thời gian cho 2 lần thử là 0.3014.**

A screenshot of a computer

Description automatically generated

**Hình b.1.3.3: Hình giao diện với thử lần 3và tốn thời gian thực thi là 2.2501 ms, bộ nhớ sử dụng là 40247296 bytes, CPU time là 9375000 ticks và tổng thời gian cho 3 lần thử là 2.5615.**

A screenshot of a computer

Description automatically generated

**Hình b.1.3.3: Hình giao diện với thử lần 3và tốn thời gian thực thi là 0.0344 ms, bộ nhớ sử dụng là 40538112 bytes, CPU time là 12187500 ticks và tổng thời gian cho 4 lần thử là 2.5959.**

**A screenshot of a computer

Description automatically generated**

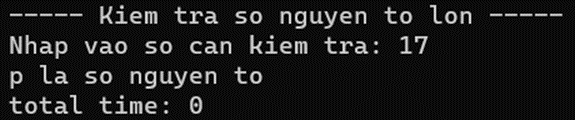
**Hình b.1.3.3: Hình thể hiện thời gian trung bình của chương trình sau 4 lần thử là 0.52 ms.**

1. Kiểm tra số nguyên tố

* Kiểm tra tính đúng đắn với số nhỏ

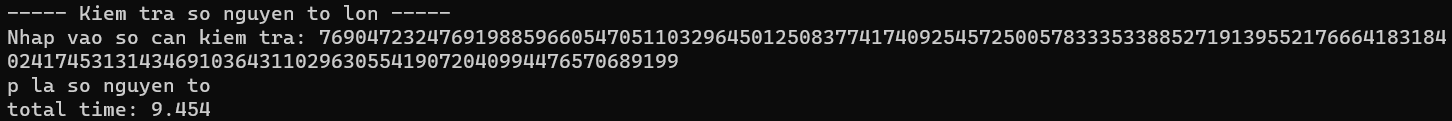
Text

Description automatically generated



* Tạo ngẫu nhiên một số nguyên tố lớn bất kì từ <https://bigprimes.org/> rồi kiểm tra thì đối với các số lớn có kích thước tầm 150 chữ số thì sẽ tốn khoảng 9.5 giây để kiểm tra.



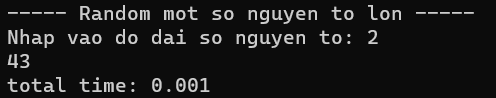


1. Tạo ngẫu nhiên số nguyên số lớn

* Thử tính đúng đắn đối với số nhỏ

Text

Description automatically generated



* Đối với các số lớn có độ dài 35 kí tự thì sẽ tốn trung bình 10 giây

Text

Description automatically generated with medium confidence

Text

Description automatically generated

A black background with white text

Description automatically generated with medium confidence

1. **Mã hóa khóa công khai RSA**
2. *Mô tả thuật toán*

* Sẽ dùng lại các hàm từ câu trước để tính

1. *Kiểm thử chương trình*

* Kiểm tra tính đúng đắn đối với số nhỏ

Text

Description automatically generated

Text

Description automatically generated

* Đối với số lớn với độ dài p và q là 70 chữ số và e là 5 chữ số thì trung bình sẽ tốn tầm 15 giây

Text

Description automatically generated

Text

Description automatically generated

1. **Ứng dụng chat có bảo mật**
2. *Mô tả hoạt động của ứng dụng chat có mã hóa bằng RSA*

Graphical user interface, website

Description automatically generated

* Chỉ được phép tạo 1 server
* Có thể tạo nhiều client
* Nhấn vào nút Listen để kết nối server.

Graphical user interface

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generated

Graphical user interface, text, application

Description automatically generated

* Khi nhập tên vào ô Client name và nhấn nút Connect thì sẽ tạo đồng thời cặp khóa của mỗi client và gửi Public key của mình cho client còn lại.
* Server cũng sẽ nhận được Public key của 2 client.

Graphical user interface, text, application, chat or text message

Description automatically generated

* Khi client soạn tin nhắn và nhấn nút Send thì tin nhắn đó sẽ được mã hóa và gửi sang client khác.
* Client kia khi nhận được tin nhắn thì tin nhắn đó đã được giải mã trong quá trình gửi từ server về.
* Một vài đoạn code quan trọng

Graphical user interface, text, application

Description automatically generated

* Server nhận được và chỉ xem được nội dung đã mã hóa, không biết nội dung chính là gì.

1. *Một số đoạn code quan trọng*

* Hàm tạo khóa:

Text

Description automatically generated

* Hàm gửi khóa từ client:

Text

Description automatically generated

* Hàm server nhận thông tin từ client:

Text

Description automatically generated

* Hàm gửi thông tin đã mã hóa từ client:

Text

Description automatically generated

* Hàm nhận thông tin đã được mã hóa đến client khác trong server và hiển thị thông tin đã được giải mã:Text

  Description automatically generated

1. **TÀI LIỆU THAM KHẢO**

[1] <https://vietcodes.github.io/algo/miller>