## **1. Giới thiệu**

**1.1 Mô tả**

Bài tập lớn này giải quyết bài toán mã hóa và giải mã văn bản bằng các thuật toán như Caesar, Rail Fence và kết hợp cả hai. Người dùng có thể nhập văn bản cần mã hóa từ file đầu vào, chọn thuật toán mã hóa và giải mã, và nhận kết quả trong file đầu ra.

**2. Hướng dẫn sử dụng**

**2.1 Cấu trúc thư mục**

- Gồm 3 foder chính là Src, Inc, Data, trong mỗi folder có các file để chạy chương trình

- Data: gồm các file đầu vào và đầu ra để chứa bản rõ và bản mã ngoài ra còn có một file từ điển ( ở đây do kích thước file từ điển lớn hơn 100 MB nên sẽ có đường link download riêng sẽ có ở cuối tệp)

* Caesar\_ciphertext.txt
* Caesar\_plaintext.txt
* Caesar\_Rail\_Fence\_ciphertext.txt
* Caesar\_Rail\_Fence\_plaintext.txt
* Rail\_Fence\_ciphertext.txt
* Rail\_Fence\_plaintext.txt
* dictionary.txt

- Inc: gồm các file header của chương trình

* Caesar\_Decrypt.h
* Caesar\_Encrypt.h
* Caesar\_Rail\_Fence\_Decrypt.h
* Caesar\_Rail\_Fence\_Encrypt.h
* Rail\_Fence\_Decrypt.h
* Rail\_Fence\_Encrypt.h

- Src: gồm các file.c của chương trình

* Caesar\_Decrypt.cpp
* Caesar\_Encrypt.cpp
* Caesar\_Rail\_Fence\_Decrypt.cpp
* Caesar\_Rail\_Fence\_Encrypt.cpp
* Rail\_Fence\_Decrypt.cpp
* Rail\_Fence\_Encrypt.cpp
* main.cpp

**2.2 Các tính năng**

Chương trình gồm 6 tính năng yêu cầu người dùng nhập số từ 1 đến 6 để lựa chọn tính năng:

1. Caesar Encryption
2. Caesar Decryption
3. Rail Fence Encryption
4. Rail Fence Decryption
5. Caesar Rail Fence Encryption
6. Caesar Rail Fence Decryption

**2.3 Cách dùng từng tính năng**

**2.3.1 Cách dùng tính năng Caesar Encryption**

- Bước 1: sử dụng câu lệnh sau để biên dịch chương trình

**g++ -fopenmp -I Inc Src/\*.cpp -o main**

- Bước 2: sử dụng câu lệnh sau để chạy chương trình

**.\main.exe**

- Bước 3: lúc này trên terminal sẽ yêu cầu người dùng lựa chọn tính năng để sử dụng, ở đây chúng ta nhập **1**

**-** Bước 4: lúc này sẽ có dòng thông báo **“ Hãy nhập bản rõ vào file Caesar\_plaintext.txt”** và **“Nhập OK để xác nhận:”**, lúc này ta sẽ nhập bản rõ vào file chương trình yêu cầu, nếu nhập xong thì nhấn **“OK”** để xác nhận, ngược lại nếu không nhập **“OK”** thì chương trình sẽ yêu cầu nhập lại với thông báo **“Vui lòng nhập OK để xác nhận:”**

- Bước 5: lúc này chương trình sẽ đọc file đầu vào nếu file không tồn tại hoặc đường dẫn sai thì sẽ xuất hiện thông báo **“Không thể mở file: Caesar\_plaintext.txt”** và chương trình sẽ dừng, ngược lại chương trình sẽ tiếp tục kiểm tra ở bước trên chúng ta có nhập gì vào file đầu vào không nếu không nhập gì thì sẽ có thông báo **“File bản rõ rỗng. Không có gì để mã hóa.”** và chương trình sẽ dừng lại, ngược lại sẽ xuất hiện thông báo “**Bản rõ đã đọc thành công từ file: Caesar\_plaintext.txt**”

- Bước 6: nhập khóa Caesar cần sử dụng

- Bước 7: lúc này chương trình sẽ bắt đầu tạo file đầu ra để ghi vào nếu không tồn tại hoặc đường dẫn sai thì xuất hiện thông báo **“Không thể tạo file: Caesar\_ciphertext.txt”**, ngược lại sẽ xuất hiện thông báo **“Bản mã đã ghi thành công vào file: Caesar\_ciphertext.txt”**, lúc này bản mã sẽ nằm trong file trong thông báo và chương trình kết thúc tại đây.

**\*** Các tính năng còn lại thực hiện tương tự như tính năng **1**

**2.4 Cấu hình hệ thống và phần mềm đang sử dụng**

- Hệ điều hành:Windows 11

- Compiler/ IDE: Visual Studio Code

- Phiên bản phần mềm: g++ 12.2.0, VS Code 1.95.3

**2.5 Thông tin liên hệ và link github**

**Địa chỉ email:** trung.lequang2002@hcmut.edu.vn

**Link download github:** <https://github.com/quangtrungcode/Assignment-Cryptography-and-Network-Security.git>

**Link download file từ điển dictionary:**

<https://drive.google.com/file/d/1RfHda4J5VEJA7En7FUegeTNtCxVTI3VS/view?usp=sharing>

### 