ĐẠI HỌC BÁCH KHOA HÀ NỘI

**Trường Công nghệ Thông tin và Truyền thông**

----- 🙡 🕮 🙣 -----



**Báo cáo Project 1**

***Đề tài:* Thuật toán mã hóa và giải mã RLC**

Giảng viên: **PGS.TS. Đặng Văn Chuyết**

Nhóm sinh viên thực hiện:

|  |  |  |
| --- | --- | --- |
| STT | Họ và tên | MSSV |
| 1 | Nguyễn Tiến Thành | 20215640 |
| 2 | Hà Văn Tăng | 20215638 |

**Hà Nội, năm 2023**

MỤC LỤC

[MỞ ĐẦU 3](#_Toc155995998)

[CHƯƠNG I. Giới thiệu bài toán 4](#_Toc155995999)

[**I.1. Thuật toán RLC – Run Length Coding** 4](#_Toc155996000)

[**I.2. Bài toán cụ thể** 4](#_Toc155996001)

[CHƯƠNG II. Phân tích bài toán 5](#_Toc155996002)

[**II.1. Thuật toán mã hóa RLC với file text không chứa chữ số** 5](#_Toc155996003)

[**II.2. Thuật toán giải mã RLC với file text không chứa chữ số** 5](#_Toc155996004)

[**II.3. Thuật toán mã hóa RLC với file text có chứa chữ số** 5](#_Toc155996005)

[**II.4. Thuật toán giải mã RLC với file text có chứa chữ số** 6](#_Toc155996006)

[CHƯƠNG III. Xây dựng chương trình 7](#_Toc155996007)

[**III.1. Hàm mã hóa RLC với file text không chứa chữ số** 7](#_Toc155996008)

[**III.2. Hàm giải mã RLC với file text được mã hóa từ file text không chứa chữ số** 9](#_Toc155996009)

[**III.3. Hàm mã hóa RLC với file text chứa chữ số** 11](#_Toc155996010)

[**III.4. Hàm giải mã RLC với file text được mã hóa từ file text có chứa chữ số** 13](#_Toc155996011)

[CHƯƠNG IV. Kiểm thử chương trình 15](#_Toc155996012)

[**IV.1. Kiểm thử chức năng mã hóa file text không chứa chữ số** 15](#_Toc155996013)

[**IV.2. Kiểm thử chức năng giải mã file text được mã hóa từ file text không chứa chữ số** 16](#_Toc155996014)

[**IV.3. Kiểm thử chức năng mã hóa file text có chứa chữ số** 17](#_Toc155996015)

[**IV.4. Kiểm thử chức năng giải mã file text được mã hóa từ file text có chứa chữ số** 18](#_Toc155996016)

[KẾT LUẬN 20](#_Toc155996017)

[PHỤ LỤC 21](#_Toc155996018)

**MỞ ĐẦU**

Ngày nay, trong bối cảnh phát triển vượt bậc của công nghệ thông tin, vấn đề về tối ưu hóa không gian lưu trữ và hiệu suất truyền là rất quan trọng. Mỗi ngày, chúng ta gặp phải lượng lớn dữ liệu cần phải được truyền tải và lưu trữ một cách hiệu quả. Để đối mặt với thách thức này, các phương pháp mã hóa dữ liệu trở nên ngày càng quan trọng. Một trong những phương pháp nén dữ liệu đơn giản nhưng hiệu quả là thuật toán mã hóa và giải mã Run-Length Encoding (RLC).

Chúng em đã phát triển một chương trình thực hiện mã hóa văn bản đầu vào thành dạng nén và sau đó giải mã nó để khôi phục lại văn bản gốc. Chương trình sẽ bắt đầu với việc mã hóa và giải mã RLC cho các bản tin chứa văn bản không có chứa số. Sau đó, chúng ta sẽ mở rộng chương trình để xử lý trường hợp bản tin có chứa số. Mục tiêu chính của dự án này là để tìm hiểu rõ về thuật toán RLC cơ bản, triển khai nó bằng ngôn ngữ lập trình C++, đánh giá hiệu suất của chương trình mã hóa và giải mã trong các tình huống khác nhau.

Chúng em hi vọng rằng báo cáo này sẽ đưa ra cái nhìn toàn diện về quá trình mã hóa và giải mã RLC sử dụng ngôn ngữ lập trình C++ và có thể hữu ích cho những người quan tâm đến lĩnh vực nén dữ liệu.

Trân trọng!

**CHƯƠNG I. Giới thiệu bài toán**

**I.1. Thuật toán RLC – Run Length Coding**

Thuật toán RLC là một phương pháp nén dữ liệu đơn giản bằng cách thay thế loạt các ký tự lặp lại bằng một ký tự và số lần lặp lại của nó. Thuật toán này giúp giảm kích thước của dữ liệu mà vẫn giữ được thông tin quan trọng.

**I.2. Bài toán cụ thể**

Thuật toán mã hóa và giải mã RLC cho một bản tin như sau:

Thuật toán mã hóa: Tìm trong bản tin những loạt tin giống hệt nhau và thay nó bằng 1 tin trong loạt và độ dài của loạt để tạo ra bản tin bị nén

Thuật toán giải mã: Tìm vị trí là độ dài của loạt và phục hồi lại loạt tin giống nhau có độ dài tìm được

Nếu trong bản tin có chứa các số thì trước vị trí của độ dài loạt của bản tin bị nén cần chèn 1 ký tự đặc biêt là ký tự DLE trong bảng mã ASCII

1. Hãy viết chương trình cho phép nhập một bản tin là đoạn văn bản không chứa các số và sử dụng thuật toán mã hóa RLC để mã hóa nó
2. Sử dụng thuật toán giải mã để phục hồi lại bản tin ban đầu từ bản tin đã bị nén bởi câu a.
3. Thực hiện lại câu a và b với đoạn văn bản có chứa số

**CHƯƠNG II. Phân tích bài toán**

**II.1. Thuật toán mã hóa RLC với file text không chứa chữ số**

**Đầu vào :** File text chứa văn bản không có chữ số

**Đầu ra :** File text chứa văn bản đã được mã hóa.

Thuật toán RLC :

* Duyệt từng dòng trong file text
* Với mỗi dòng, kiểm tra xem có chứa chữ số hay không. Nếu có thì dừng lại và thông báo file có chứa chữa số. Ngược lại thì làm bước tiếp theo
* Duyệt qua từng ký tự của dòng đó để xác định các loạt tin giống nhau, xác định độ dài của loạt tin
* Nếu độ dài loạt tin lớn hơn 2 thì thay thế loạt tin bằng độ dài của loạt tin và ký tự bị lặp lại. Ngược lại thì giữ nguyên.
* Lưu kết quả thu được vào file text đầu ra chính là file mã hóa
* Tiến hành tính toán kích thước file gốc, kích thước file giải mã, tỷ lệ nén file và thời gian thực thi.

## **II.2. Thuật toán giải mã RLC với file text không chứa chữ số**

**Đầu vào :** File text chứa văn bản đã được nén từ một văn bản không chứa số

**Đầu ra :** File text chứa văn bản đã được giải mã

Thuật toán RLC :

* Duyệt từng dòng trong file text
* Với mỗi dòng, duyệt qua từng ký tự của dòng đó để xác định được độ dài của loạt tin đã được mã hóa và ký tự được mã hóa. Tiến hành khôi phục lại loạt tin.
* Lưu kết quả thu được và file text đầu ra chính là file giải mã
* Tiến hành tính toán kích thước file gốc, kích thước file giải mã, tỷ lệ giải nén file và thời gian thực thi.

**II.3. Thuật toán mã hóa RLC với file text có chứa chữ số**

**Đầu vào :** File text chứa văn bản có chữ số

**Đầu ra :** File text chứa văn bản đã được mã hóa.

Thuật toán RLC :

* Duyệt từng dòng trong file text
* Duyệt qua từng ký tự của dòng đó để xác định các loạt tin giống nhau, xác định độ dài của loạt tin, với quy ước độ dài loạt tin nhỏ hơn 10. Khi độ dài đạt tới 9 thì cần thực hiện xử lý luôn.
* Nếu độ dài loạt tin lớn hơn 2 thì thay thế loạt tin bằng xâu kết hợp giữa ký tự dle, độ dài của loạt tin và ký tự bị lặp lại, tiếp tục duyệt dòng đó từ vị trí hiện tại. Ngược lại thì giữ nguyên.
* Lưu kết quả thu được vào file text đầu ra chính là file mã hóa
* Tiến hành tính toán kích thước file gốc, kích thước file giải mã, tỷ lệ nén file và thời gian thực thi.

**II.4. Thuật toán giải mã RLC với file text có chứa chữ số**

**Đầu vào :** File text chứa văn bản đã được nén từ một văn bản có chứa số

**Đầu ra :** File text chứa văn bản đã được giải mã

Thuật toán RLC :

* Duyệt từng dòng trong file text
* Với mỗi dòng, duyệt qua từng ký tự của dòng đó để xác định được độ dài của loạt tin đã được mã hóa và ký tự được mã hóa. Độ dài loạt tin có giá trị từ 3 đến 9, luôn đứng sau ký tự dle và đứng trước ký tự được mã hóa. Tiến hành khôi phục lại loạt tin.
* Lưu kết quả thu được và file text đầu ra chính là file giải mã
* Tiến hành tính toán kích thước file gốc, kích thước file giải mã, tỷ lệ giải nén file và thời gian thực thi.

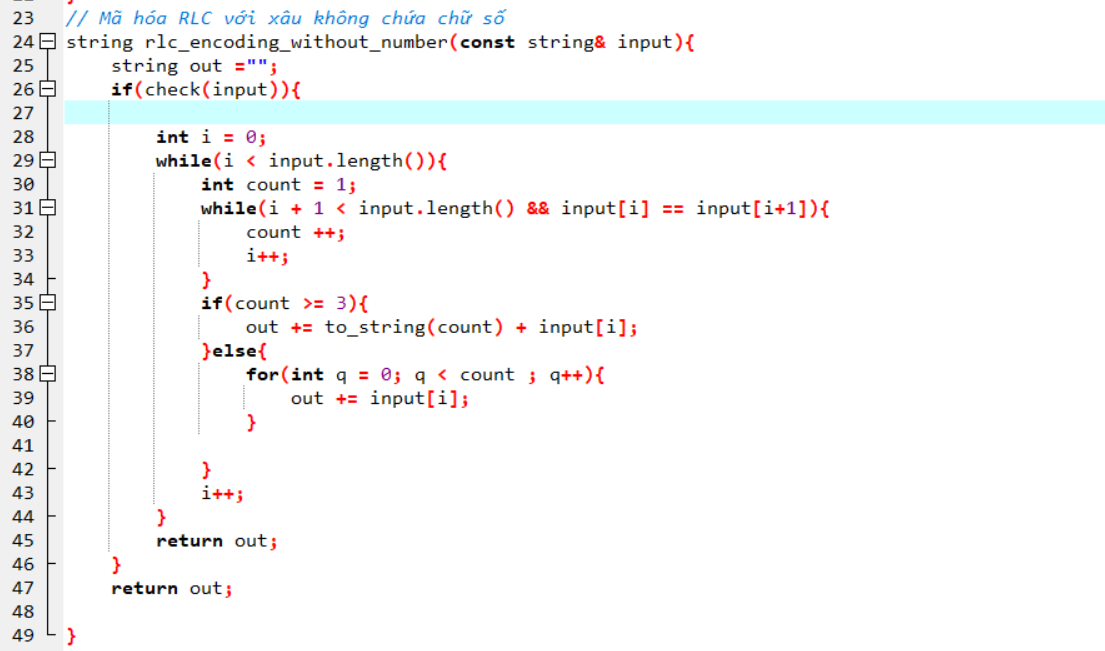
# **CHƯƠNG III. Xây dựng chương trình**

Chương trình mã hóa và giải mã RLC được xây dựng để thực hiện quá trình nén và giải nén dữ liệu văn bản với bốn chức năng chính :

1. Mã hóa RLC với file text không chứa chữ số
2. Giải mã RLC với file text được mã hóa từ file text không chứa chữ số
3. Mã hóa RLC với file text có chứa chữ số
4. Giải mã RLC với file text được mã hóa từ file text có chứa chữ số

Chương trình cũng cung cấp Menu linh hoạt cho phép người dùng lựa chọn sử dụng các chức năng trên, cũng như các thông tin bên cạnh như kích thước các file, hiệu suất, thời gian thực thi.

## **III.1. Hàm mã hóa RLC với file text không chứa chữ số**

****

*Hàm mã hóa RLC với xâu không chứa chữ số*

A screenshot of a computer program

Description automatically generated

A computer code with text

Description automatically generatedA close-up of a white background

Description automatically generated

*Hàm mã hóa RLC với file text không chứa chữ số*

## **III.2. Hàm giải mã RLC với file text được mã hóa từ file text không chứa chữ số**

**A computer screen shot of a code

Description automatically generated**

*Hàm giải mã RLC với văn bản được mã hóa từ văn bản không chứa số*

**A screenshot of a computer program

Description automatically generated**

**A screenshot of a computer code

Description automatically generated**

*Hàm giải mã RLC với file text được mã hóa từ file text không chứa số*

## **III.3. Hàm mã hóa RLC với file text chứa chữ số**

**A screenshot of a computer program

Description automatically generated**

*Hàm mã hóa RLC với văn bản có chứa chữ số*

**A screenshot of a computer program

Description automatically generated**

**A computer screen shot of text

Description automatically generated**

*Hàm mã hóa RLC với file text có chứa chữ số*

## **III.4. Hàm giải mã RLC với file text được mã hóa từ file text có chứa chữ số**

**A screenshot of a computer program

Description automatically generated**

*Hàm giải mã RLC với văn bản được mã hóa từ văn bản có chứa chữ số*

**A screenshot of a computer program

Description automatically generated**

**A computer screen shot of a program

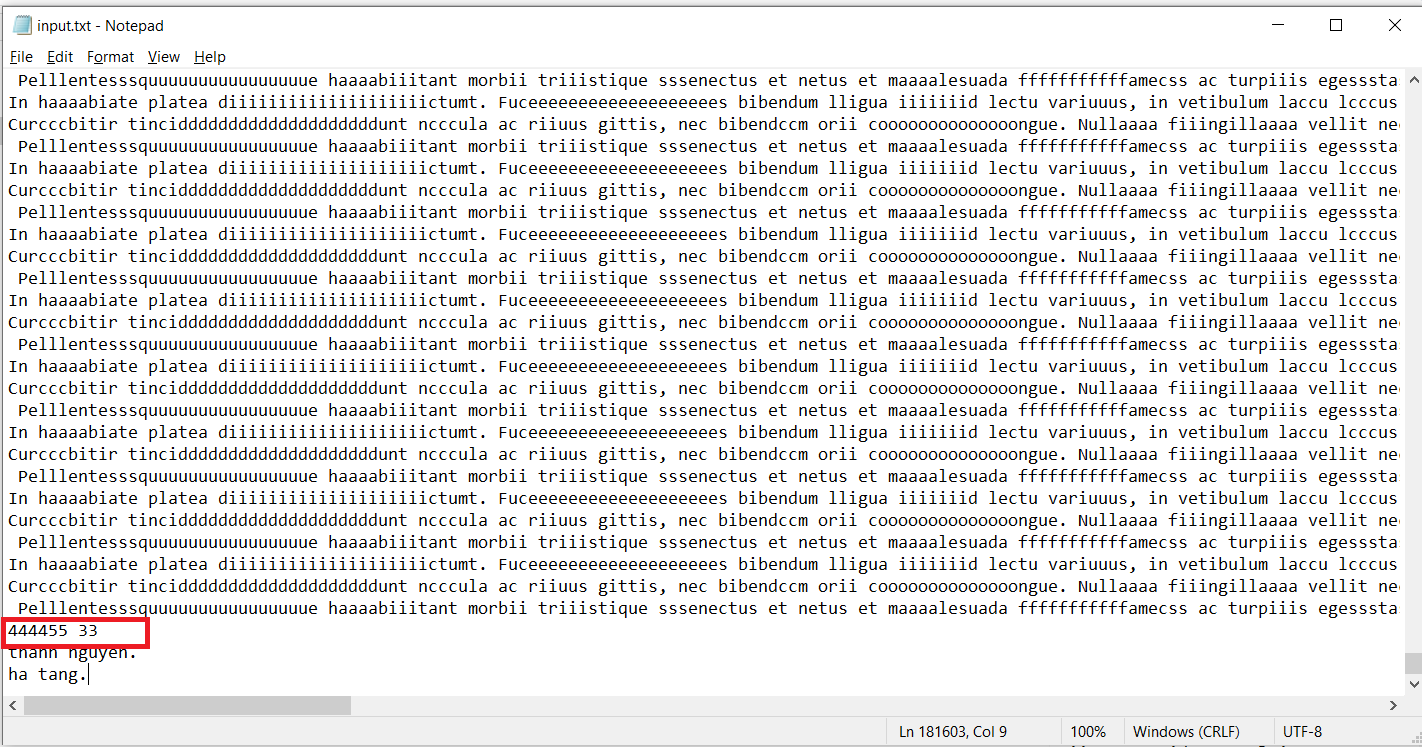
Description automatically generated**

*Hàm giải mã RLC với file text được mã hóa từ file có chứa chữ số*

# **CHƯƠNG IV. Kiểm thử chương trình**

## **IV.1. Kiểm thử chức năng mã hóa file text không chứa chữ số**

-Trường hợp 1: file input.txt có chứa chữ số

****

**A screen shot of a computer

Description automatically generated**

-Trường hợp 2: file input.txt không chứa chữ số

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

## **IV.2. Kiểm thử chức năng giải mã file text được mã hóa từ file text không chứa chữ số**

**A screenshot of a computer screen

Description automatically generated**

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

## **IV.3. Kiểm thử chức năng mã hóa file text có chứa chữ số**

****

**A screenshot of a computer

Description automatically generated**

****

## **IV.4. Kiểm thử chức năng giải mã file text được mã hóa từ file text có chứa chữ số**

**A screen shot of a computer

Description automatically generated**

A screenshot of a computer program

Description automatically generated

**A screenshot of a computer screen

Description automatically generated**

# **KẾT LUẬN**

Như vậy, chương trình mã hóa RLC mà chúng em phát triển đã thành công trong việc thực hiện các chức năng, nhiệm vụ mã hóa và giải mã dữ liệu dạng văn bản dựa trên thuật toán RLC – Run-Length Coding. Chương trình có khả năng xử lý và nén giải dữ liệu đáng kể khi có sự lặp lại liên tiếp của các ký tự trong file đầu vào, cũng như cung cấp chức năng giải mã file về dạng ban đầu.

Qua đây, ta hiểu được một cách tổng quan, cơ bản về phương thức hoạt động của thuật toán mã hóa RLC, thấy được sự hiệu quả của thuật trong việc giảm kích thước dữ liệu dạng văn bản cũng như tiềm năng cải tiến trong tương lai.

# **PHỤ LỤC**

Mã nguồn của chương trình : <https://github.com/thanhf47/project1>