## **ĐẠI HỌC QUỐC GIA TP.HCM**

## **ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

## KHOA CÔNG NGHỆ THÔNG TIN





## BÁO CÁO ĐỒ ÁN

Môn: Cấu trúc dữ liệu và giải thuật

1. **Thông tin nhóm:**

|  |  |  |
| --- | --- | --- |
| **Họ và tên** | **MSSV** | **Gmail** |
| Võ Trọng Gia Vinh | 18120652 | vgia1055@gmail.com |
| Lê Thành Việt | 18120648 | lethanhviet7c@gmail.com |

**II. Mức độ hoàn thành:**

***1. Những câu đã làm được:***

* + Nén 1 tập tin chuỗi.
  + Nén 1 thư mục chứa nhiều tập tin chuỗi.
  + Nén 1 tập tin có các kiểu dữ liệu còn lại.
  + Nén 1 thư mục có chứa nhiều tập tin với nhiều định dạng.

***2. Những câu chưa làm được:*** 0

**III. Thuật toán nén được áp dụng:**

Thuật toán nén được sử dụng trong đồ án là **Thuật toán nén Huffman**. Thuật toán dựa trên tần suất xuất hiện của các ký tự trong dữ liệu cần nén để phát sinh mã bit thay thế sao cho mã bit thay thế là ngắn nhất.

***Thuật toán Huffman:***

Thuật toán được nhóm áp dụng vào đồ án qua 4 bước:

Bước 1: Duyệt dữ liệu để thống kê tần số xuất hiện của các ký tự.

Bước 2: Tạo cây Huffman từ bảng thống kê.

Bước 3: Phát sinh mã bit thay thế cho các ký tự.

Bược 4: Lưu thông tin cây Huffman và thay thế những ký tự của dữ liệu bằng những mã bit thay thế đã được phát sinh ở bước 3.

***Cây Huffman:***

* Là 1 cây nhị phân đầy đủ.
* Các nút có tần số cao nằm gần gốc hơn
* Tổng số nút là 2n – 1 với n là số ký tự khác nhau có trong dữ liệu

***Các nút của cây Huffman:***

* Mỗi nút sẽ gồm 1 ký tự và trọng số.
* Mỗi nút lá sẽ chứa 1 ký tự trong dữ liệu
* Trọng số của nút lá là số lần xuất hiện của ký tự ở nút lá trong dữ liệu.
* Nút cha có trọng số bằng tổng trọng số của hai nút con
* Nút con bên trái có trọng số nhỏ hơn hoặc bằng nút con bên phải

***Cách tạo cây Huffman:***

Bước 1: Chọn ra hai phần tử có trọng số thấp nhất a và b.

Bước 2: Tạo 1 ký tự c có trọng số bằng tổng trọng số của a và b.

Bước 3: Loại bỏ a và b ra khỏi bảng thống kê, thêm c vào bảng thống kê.

Bước 4: Lặp lại các bước trên cho đến khi bảng chỉ còn lại 1 phần tử.

***Những hạn chế của thuật toán:***

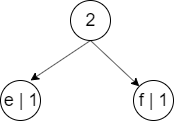
1. Duyệt dữ liệu 2 lần ( 1 lần thống kê, 1 lần mã hóa) → chi phí cao nếu dữ liệu lớn.
2. Phải lưu trữ lại thông tin của cây Huffman để giải nén → tăng kích thước dữ liệu nén.

***Ví dụ minh họa cách áp dụng thuật toán:***

* File dữ liệu có chưa nội dung là một đoạn text : aaabbbbcccddef
* Duyệt file dữ liệu để lấy tần số các ký tự:

|  |  |
| --- | --- |
| Ký tự | Tần số |
| a | 3 |
| b | 4 |
| c | 3 |
| d | 2 |
| e | 1 |
| f | 1 |

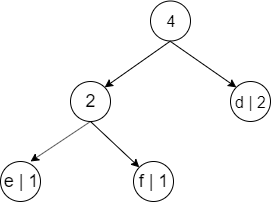
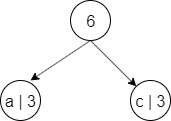
* Chọn hai ký tự có tần số nhỏ nhất gộp thành một:

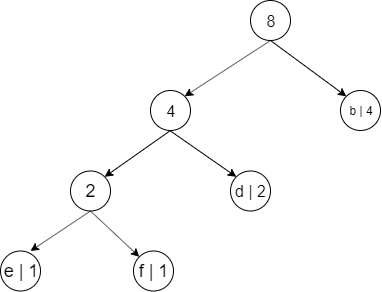


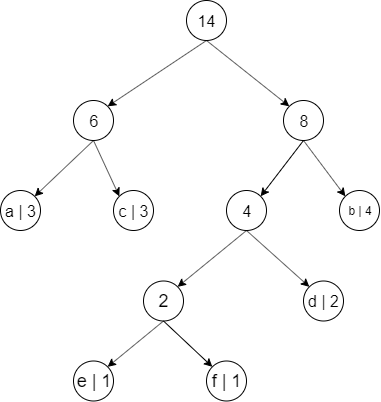
* Xóa bỏ hai ký tự vừa chọn và thêm ký tự mới vào tần số:

|  |  |
| --- | --- |
| Ký tự | Tần số |
| a | 3 |
| b | 4 |
| c | 3 |
| d | 2 |
| INVALID | 2 |

* Lặp lại tương tự các bước trên ta sẽ xây dựng được cây Huffman:





* Dữ liệu mã hóa:

|  |  |
| --- | --- |
| Ký tự | Mã bit |
| a | 00 |
| b | 11 |
| c | 01 |
| d | 101 |
| e | 1000 |
| f | 1001 |

* Kết quả chuỗi ban đầu sẽ được mã hóa thành:

0000001111111101010110110110001001

**IV. Công việc của mỗi thành viên:**

1. **Lê Thành Việt:**

* Huffman nén file :
* Duyệt file dữ liệu và tạo bảng tần số :

void getSymbolsFromFile();

* Sắp xếp các node cây Huffman theo tần số tăng dần :

void sortSymbol(vector<node\*> &tree);

* Khởi tạo cây Huffman :

void creatHuffmanTree();

* Lấy chuỗi bit mã hóa cho các ký tự :

string getPathToLeaf(node\* crr, char symbol, string path);

* Nén dữ liệu:

void encode();

1. **Võ Trọng Gia Vinh:**

* Nén folder :
* Duyệt folder và tạo đường dẫn đến các file con:

void ListFileInFolder(vector<string>& list\_file, string path\_folder, string path\_header);

* Lưu cây Huffman vào file nén:

bool restoreTree(node\* root, string& result);

* Tạo lại cây Huffman từ file nén:

bool rebuildTree(node\*& root, string& code);

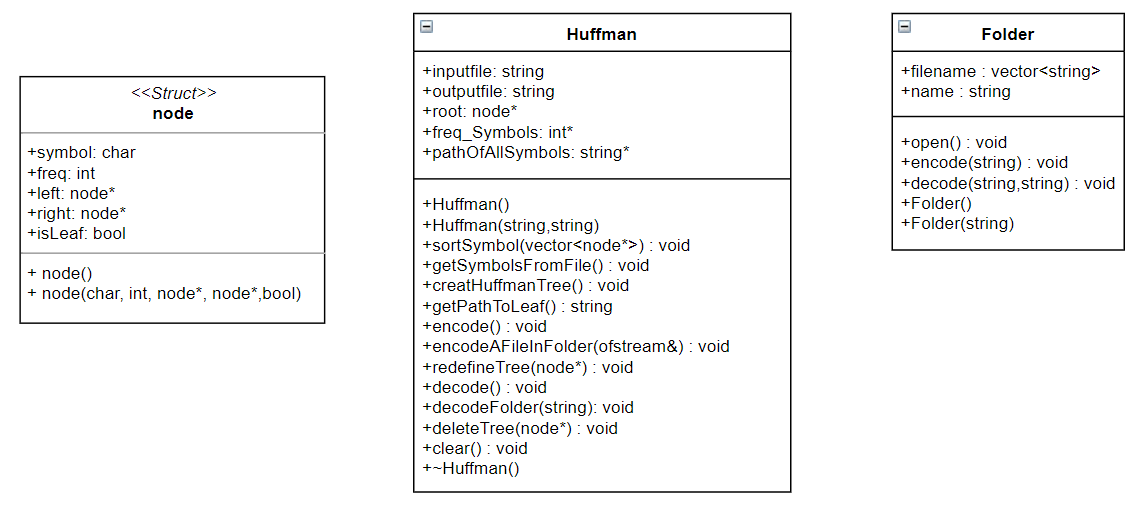
* Nén folder, tổ chức bố cục trong file nén:

void encode(string outputfile);

* Giải nén folder/file:

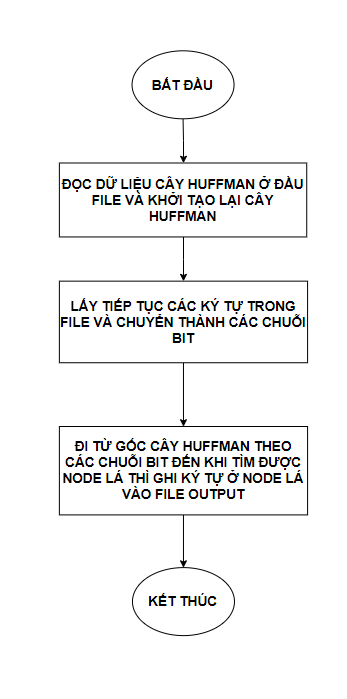
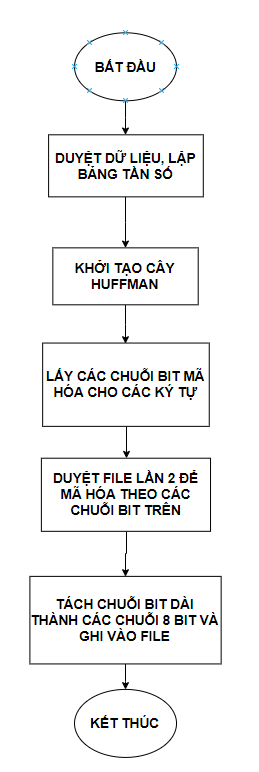
void Folder::decode(string inputfile, string outfolder);

**V. Sơ đồ lớp:**

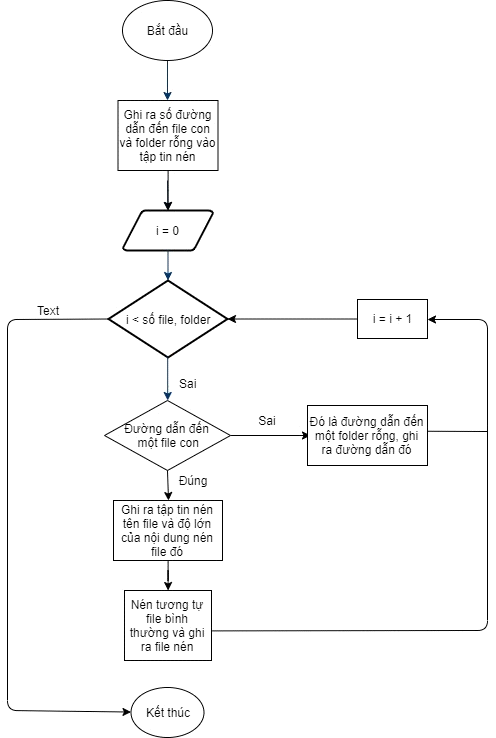


**VI. Lưu đồ thuật toán:**

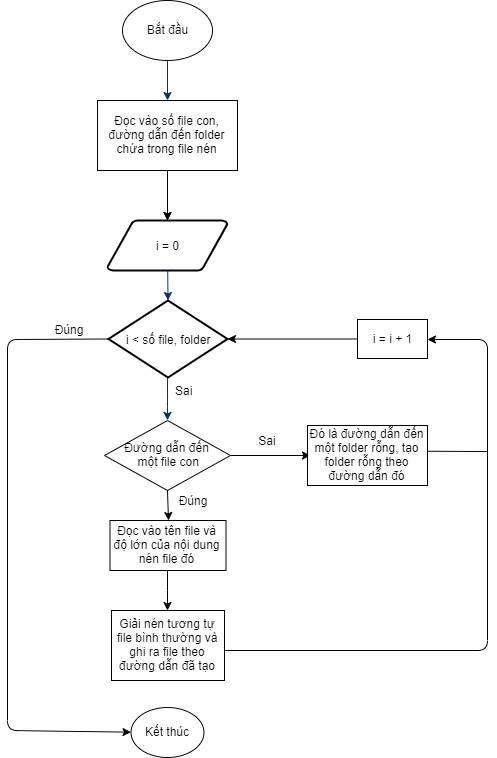
1. **Nén và giải nén file:**

****

1. **Nén Folder:**

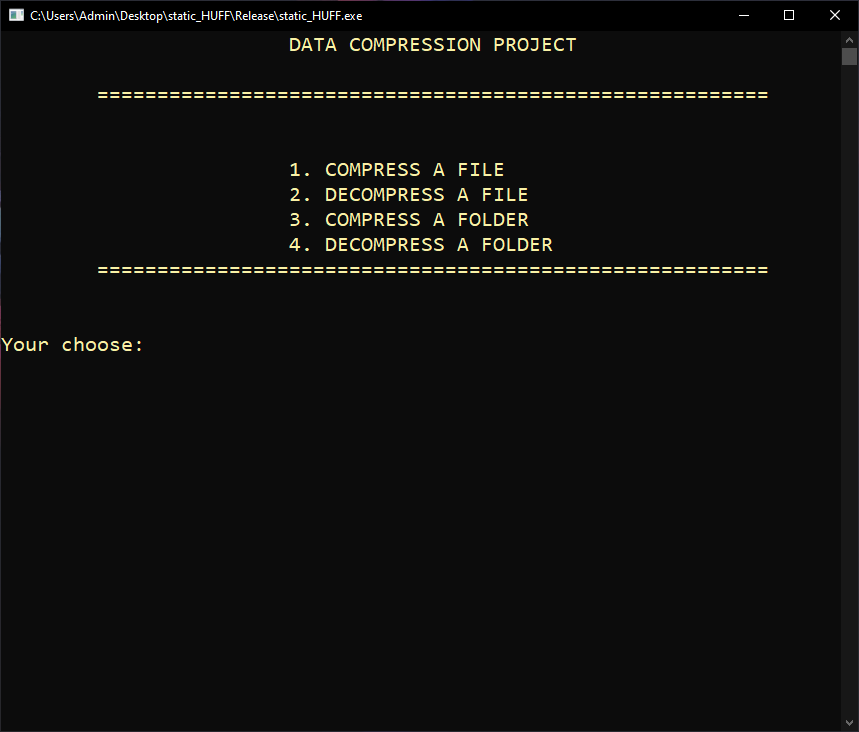


1. **Giải nén Folder:**



**VII. Hướng dẫn sử dụng:**

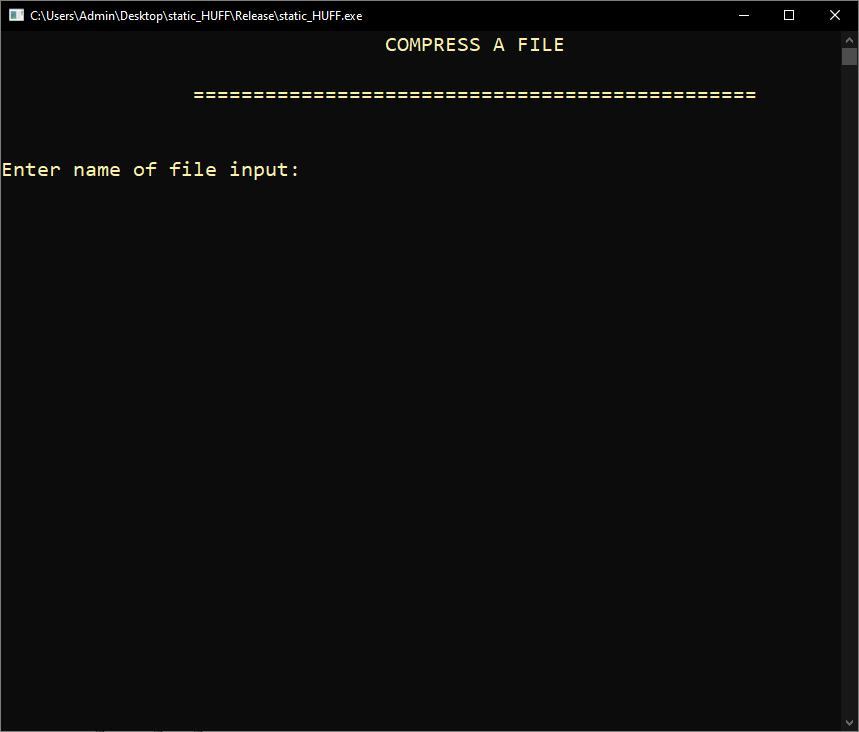
Giao diện chính của chương trình gồm có 4 phần như hình bên dưới:



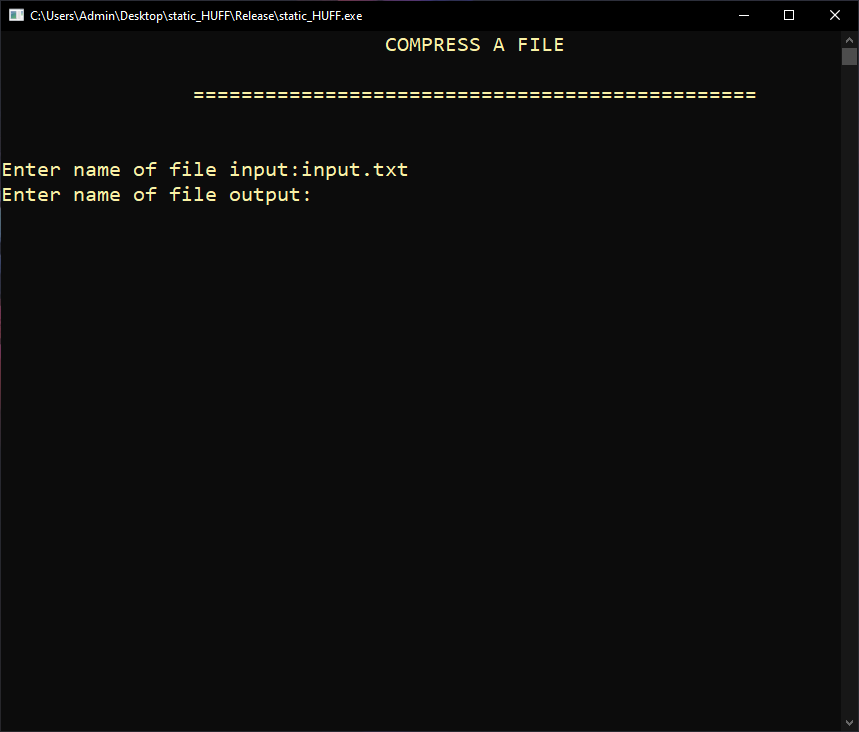
Người dùng chọn chức năng tương ứng bằng cách nhập các số 1,2,3,4 vào Your choose:

Sau khi nhập nhấn Enter, chương trình sẽ chuyển sang giao diện mới của chức năng tương ứng

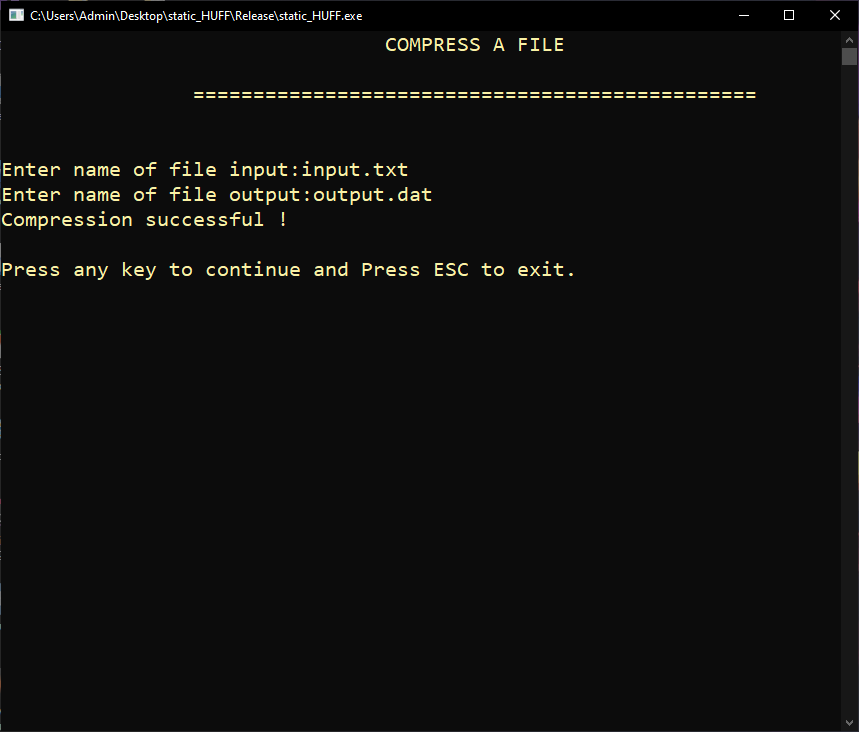
1. ***Nén 1 file dữ liệu:***



Người dùng nhập tên file cần nén và nhấn Enter.

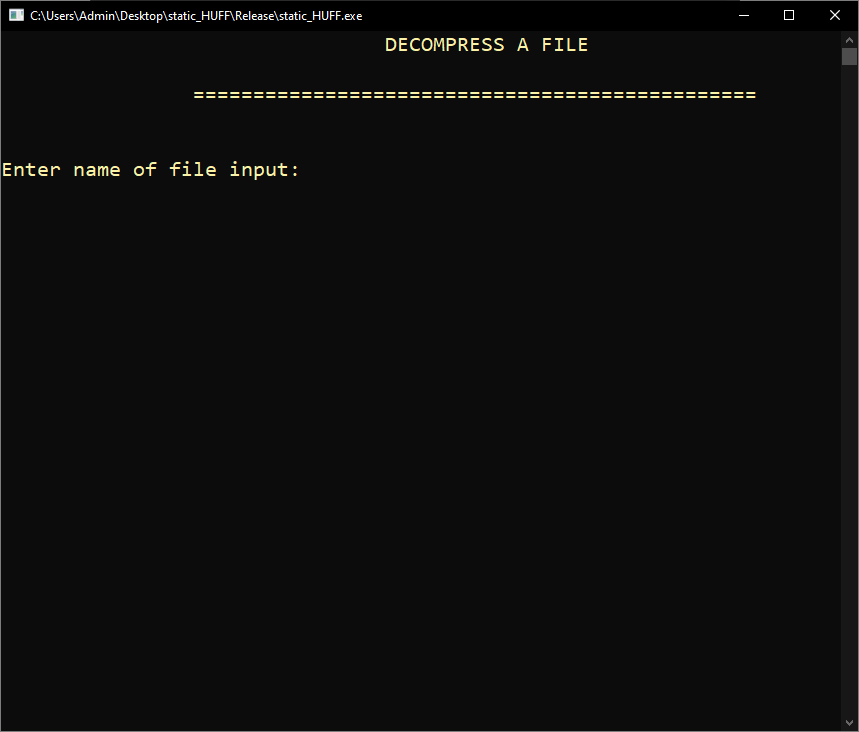


Sau đó nhập tên file output và nhấn Enter.

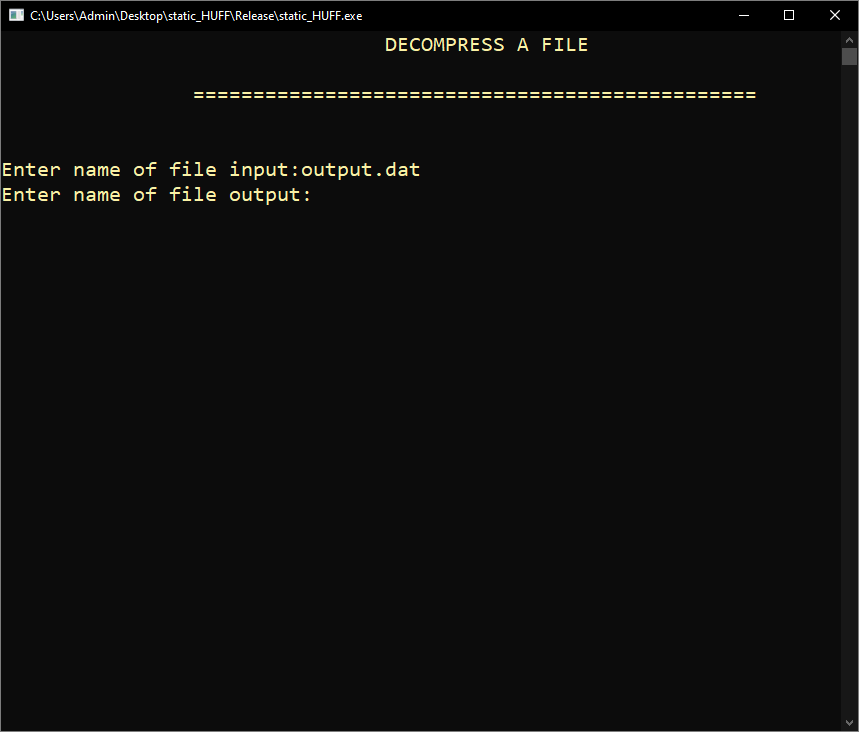


Nén thành công file input.txt thành file output.dat

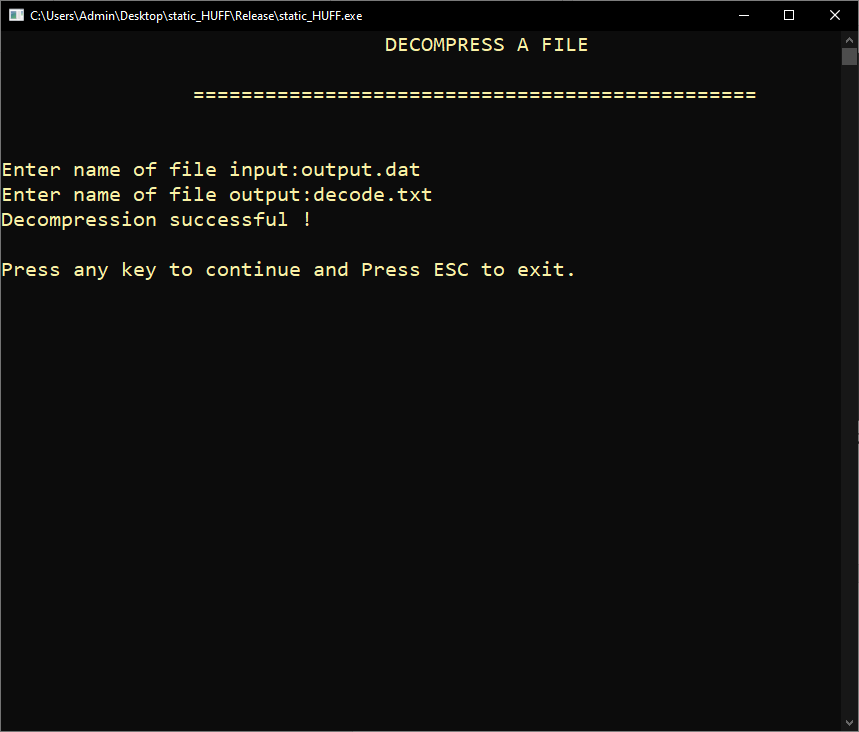
1. ***Giải nén file:***



Nhập tên file input là file dữ liệu đã được nén và nhấn Enter.



Nhập tên file output là file sẽ được giải nén ra và nhấn Enter.

\

Giải nén thành công !

1. ***Nén folder chứ nhiều file:***



Nhập tên folder cần nén và nhấn Enter.



Nhập tên file output và nhấn Enter.

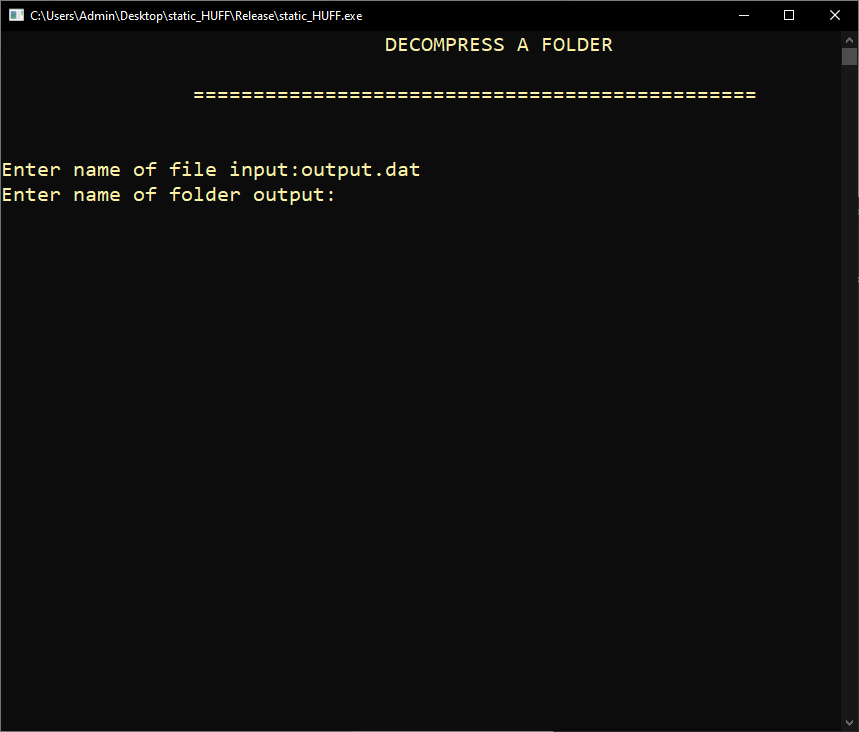


Nén thành công folder.

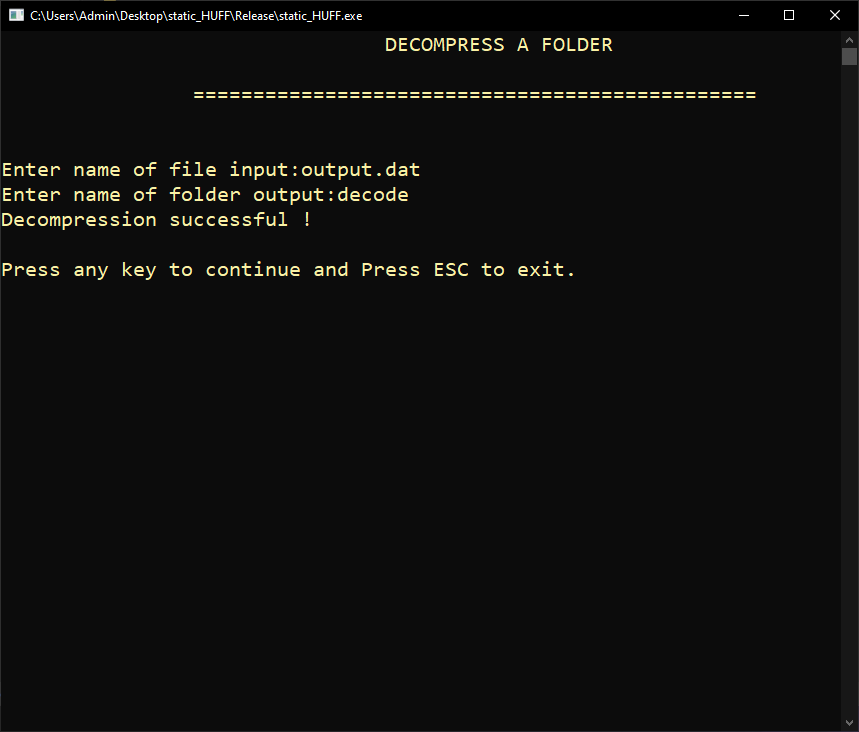
1. Giải nén folder:



Nhập tên file nén của folder đã nén và nhấn Enter.



Nhập tên folder mới là folder sẽ được giải nén ra và nhấn Enter.



Giải nén folder thành công.

**VIII. Video hướng dẫn sử dụng:**

<https://www.youtube.com/watch?v=BGP45qZ_0O4&t=81s>