|  |  |
| --- | --- |
|  |  |
| TRƯỜNG ĐẠI HỌC CÔNG NGHỆ SÀI GÒN  **KHOA CÔNG NGHỆ THÔNG TIN**  ---oOo--- | |
|  | |
|  | |
| **Bài tập nhóm Mã Hóa Ứng Dụng** | |
|  | |
| *Tên đề tài:*  **Mã hóa cổ điển** | |
| Giáo viên hướng dẫn : **Đoàn Trình Dục**  Sinh viên thực hiện : **D20\_TH08**  **Biện Hồng Phát (nhóm trưởng)(Dh52006010)**  **Trần Trọng Phát (DH52006015) Nguyễn Hồng Gia Huy(DH52005847)** | |
|  | |
| TP HỒ CHÍ MINH - NĂM 2023 | |

**NHẬN XÉT**

**(Của Giáo viên)**

**MỤC LỤC**

NHẬN XÉT Trang 2

MỤC LỤC Trang 3

Chương 1: TỔNG QUAN VỀ THUẬT TOÁN Trang 5

* 1. Lý thuyết về Thuật toán Trang 5

Chương 2: THIẾT KẾ THUẬT TOÁN Trang 5

* 1. Công cụ thiết kế Trang 5
  2. Thiết kế giao diện Trang 6
  3. Các chức năng Trang 8

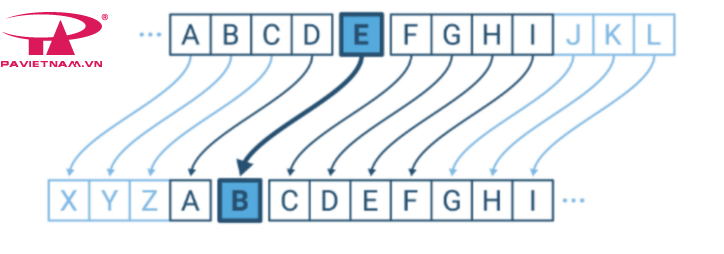
Chương 3: CÁCH SỬ DỤNG CHƯƠNG TRÌNH VÀ KẾT QUẢ Trang 11

3.1. Kết quả và cách sử dụng Trang 11

Chương 1. **TỔNG QUAN VỀ THUẬT TOÁN**

* 1. **Lý thuyết về thuật toán**

Mã hoá cổ điển là phương pháp mã hóa đơn giản nhất, tồn lại lâu nhất trên thế giới và không cần khóa bảo mật, chỉ cần người gửi và người nhận cùng biết về thuật toán này là được.



Ví dụ: nếu chúng ta dùng thuật toán đổi kí tự trong câu văn thành kí tự liền kề trong bảng chữ cái thì chữ “PAvietnam” sẽ biến thành “QBxkfuobn”. Người nhận khi nhận được chữ “QBxkfuobn” thì chỉ việc dịch ngược lại là xong.

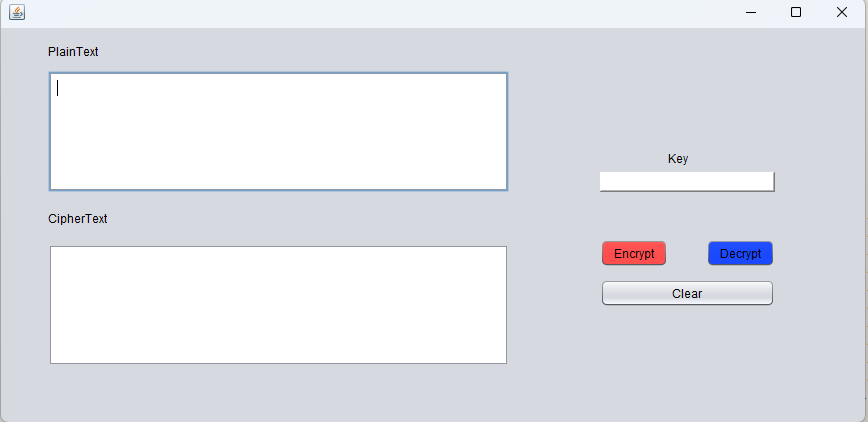
Tuy nhiên, phương pháp mã hóa này được xem là không an toàn, vì nếu một người thứ ba biết được thuật toán thì xem như thông tin không còn bảo mật nữa. Việc giữ bí mật thuật toán trở nên rất quan trọng, và không phải ai cũng có thể giữ bí mật đó một cách trọn vẹn. Có khả năng người đó sẽ rò rỉ ra, hoặc có ai đó ngồi giải ra thuật toán thì xem như chúng ta thất bại.

Chương 2. **THIẾT KẾ THUẬT TOÁN**

* 1. **Công cụ thiết kế**
* Sử dụng Netbean làm IDE
* Ngôn ngữ :Java
* Plugin Swing:Dùng để lập trình thiết kế giao diện
  1. **Thiết kế giao diện**

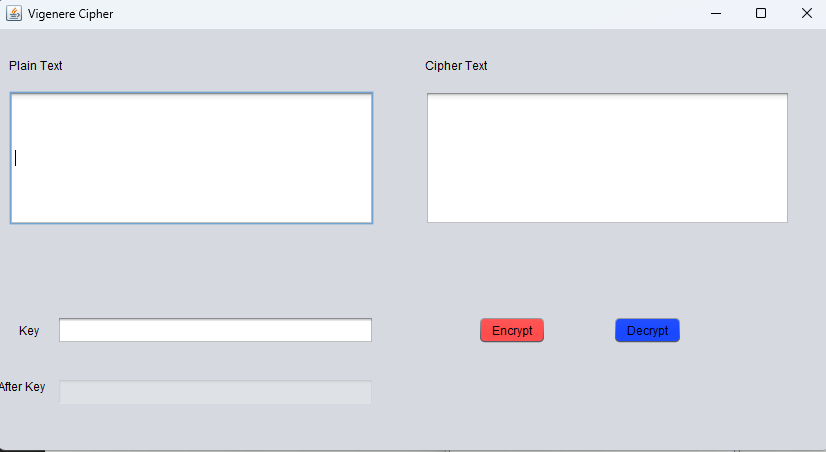
**Ceasar**

Đây là giao diện



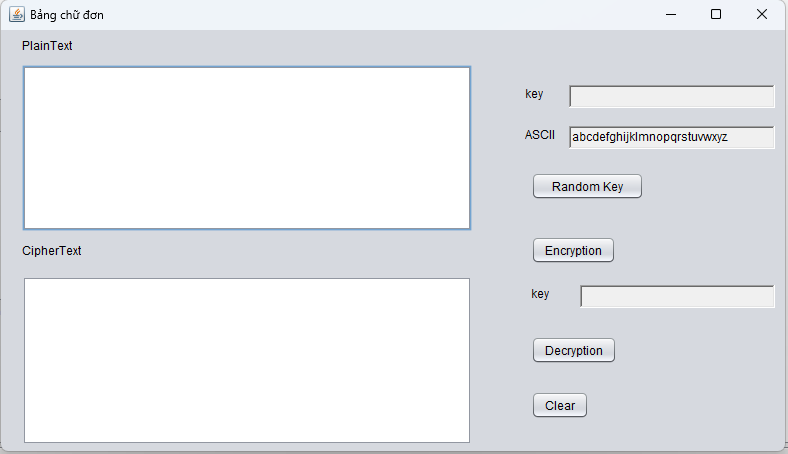
**Vegenere**

Đây là giao diện

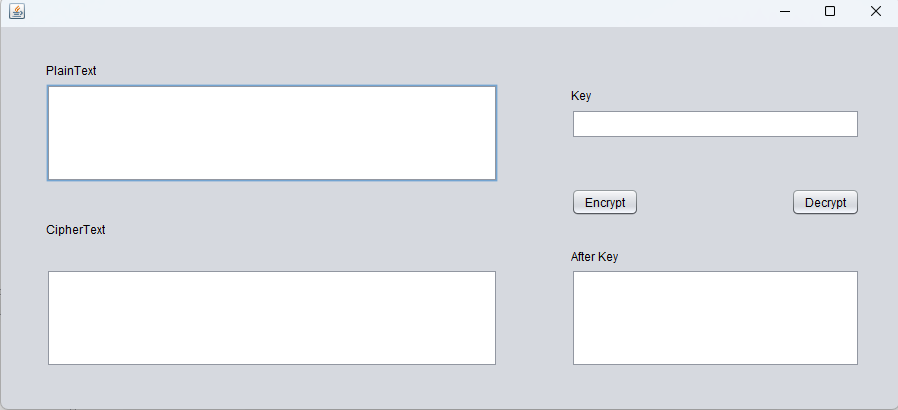


**Bảng chữ cái đơn ( Single Alphabet)**

Đây là giao diện ( được căn giữa màn hình)

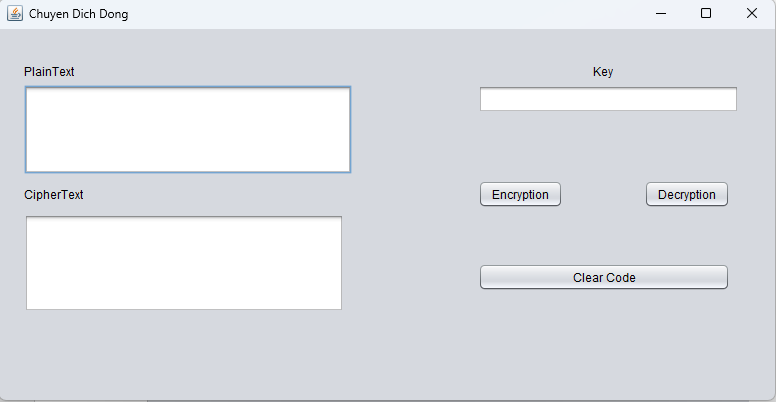


**PlayFair**Đây là giao diện

****

**Chuyển dịch dòng**

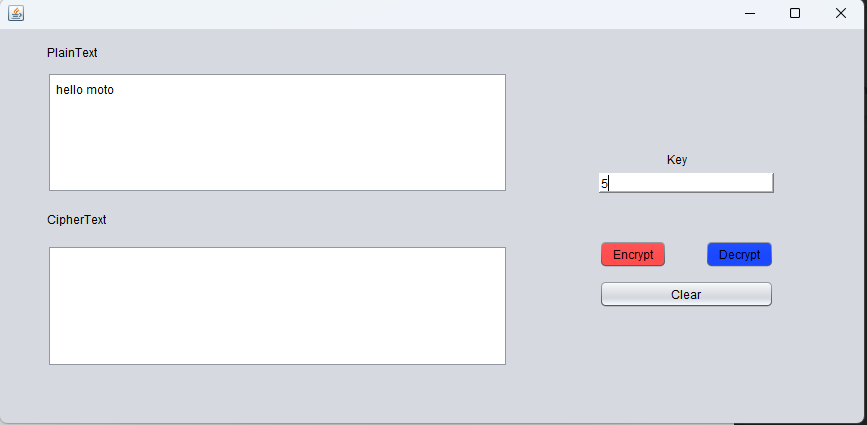
Đây là giao diện



* 1. **Các chức năng**

**Ceasar**

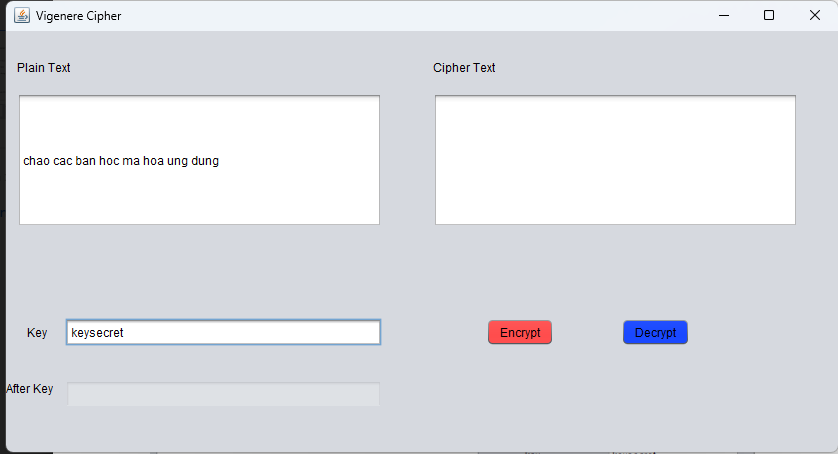
Với những tính năng như sau:



* PlainText: Văn bản cần mã hóa
* CipherText: Văn bản sau khi mã hóa  
  Key: Khóa để mã hóa
* Encrypt: Nút để thực hiện mã hóa
* Decrypt: Nút để thực hiện giải mã
* Clear : Nút làm sạch chương trình (plaintext + ciphertext)

**Vegenere**

Với các tính năng:



Plaintext, Ciphertext, Key, Encrypt , Decrypt tương tự như trên

After Key: văn bản sau khi mã hóa

**Bảng chữ cái đơn ( Single Alphabet)**

Các tính năng:



PlainText, CipherText, key, Encrypt, Decrypt tương tự như trên

ASCII : bảng mã ascii

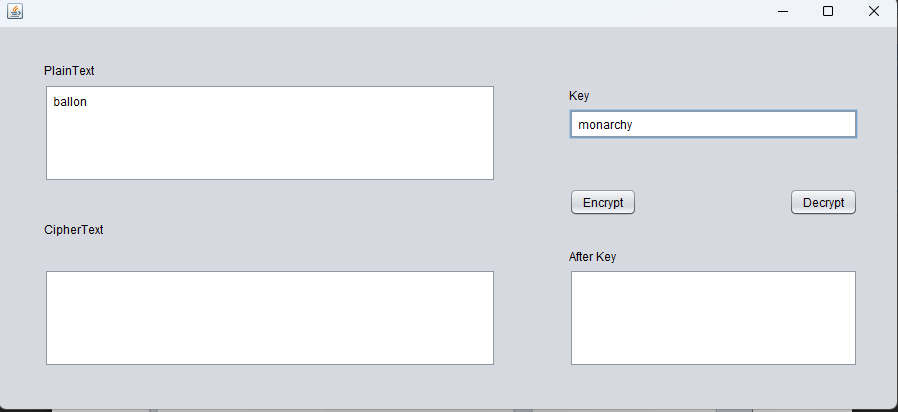
Random Key: nút random key để mã hóa

Clear: Dọn sạch màn hình

**\*\*\*** key và mã ascii đã được disable **\*\*\***

**PlayFair**

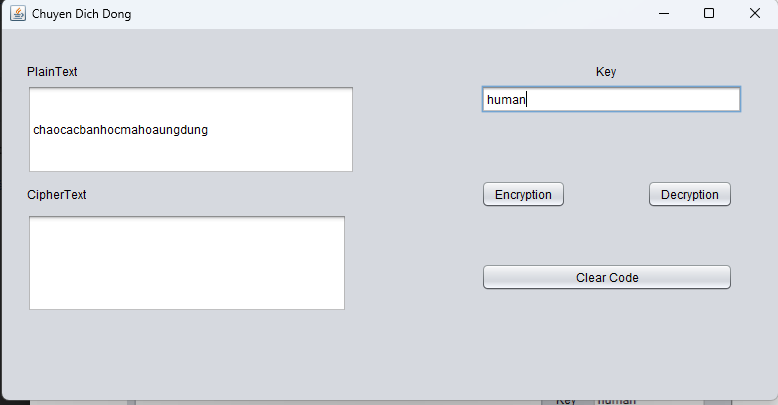
Với các tính năng



Tương tự trên với after key là key sẽ nhận được sau khi mã hóa

**Chuyển dịch dòng**

Với các tính năng



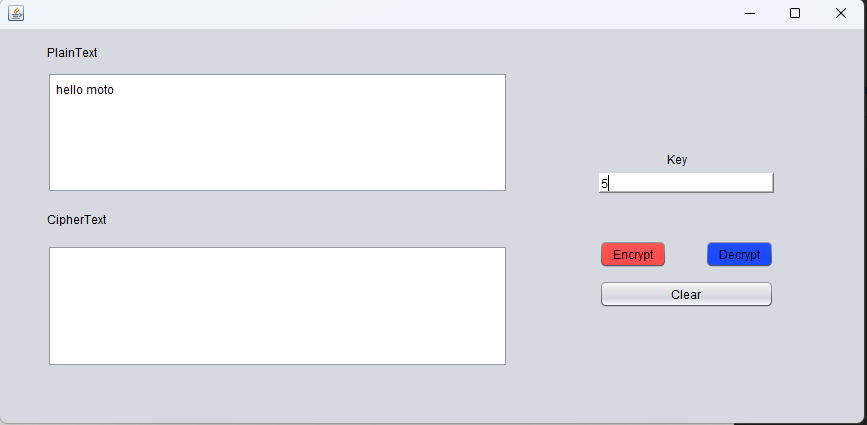
Tất cả tính năng đều tương tư như trên

Chương 3. **CÁCH** **SỬ DỤNG CHƯƠNG TRÌNH VÀ KẾT QUẢ**

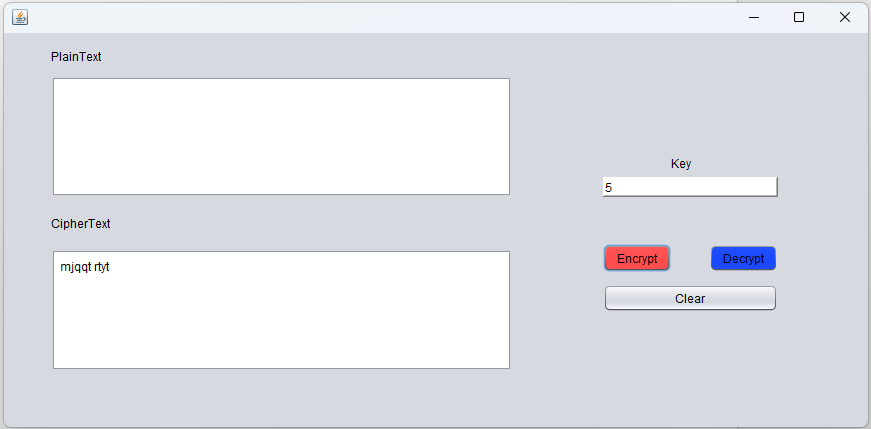
**3.1 Kết quả và cách sử dụng**

**Ceasar**

Khi ta nhập dữ liệu và bấm nút encrypt như hình dưới :

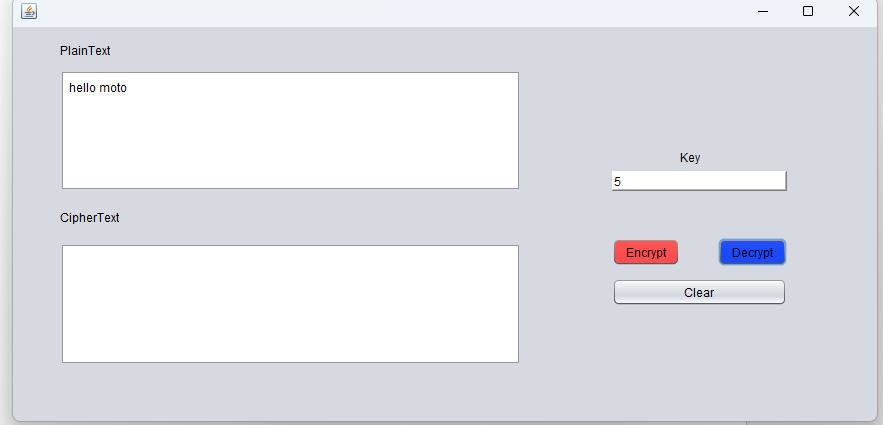


Thì kết quả trả ra được như sau:



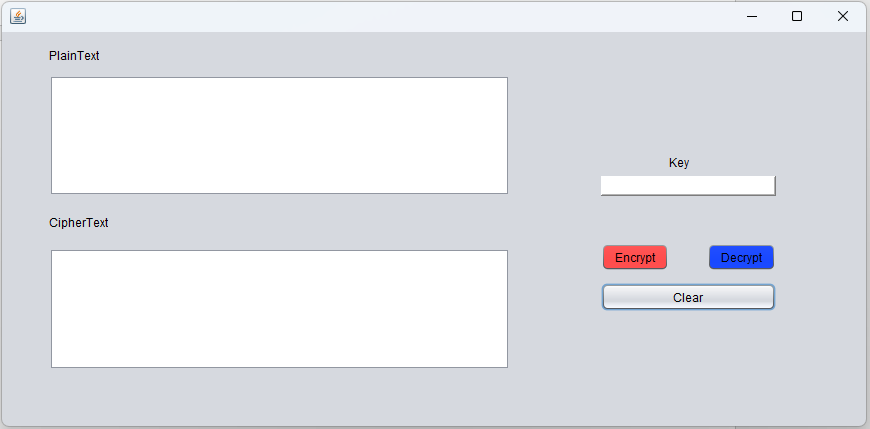
Đoạn văn bản được mã hóa.

Khi bấm nút Decrypt :



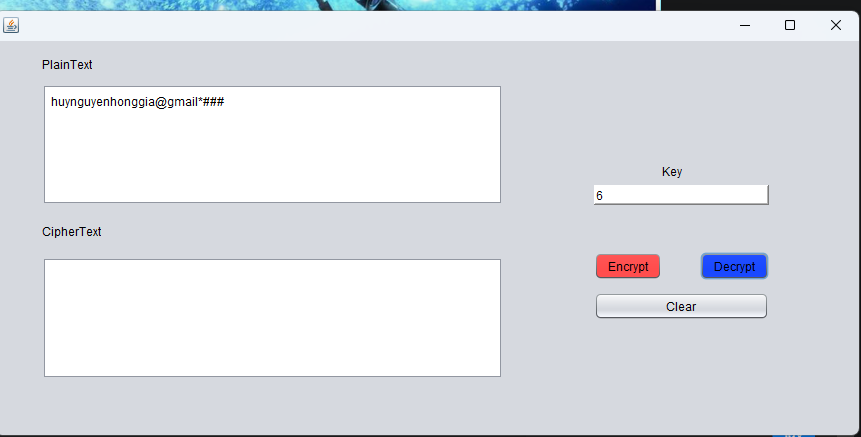
Đoạn mật mã đã được giải mã và khi bấm nút encrypt và decrypt kết quả vẫn sẽ ra giống nhau.

Khi ta bấm nút Clear:

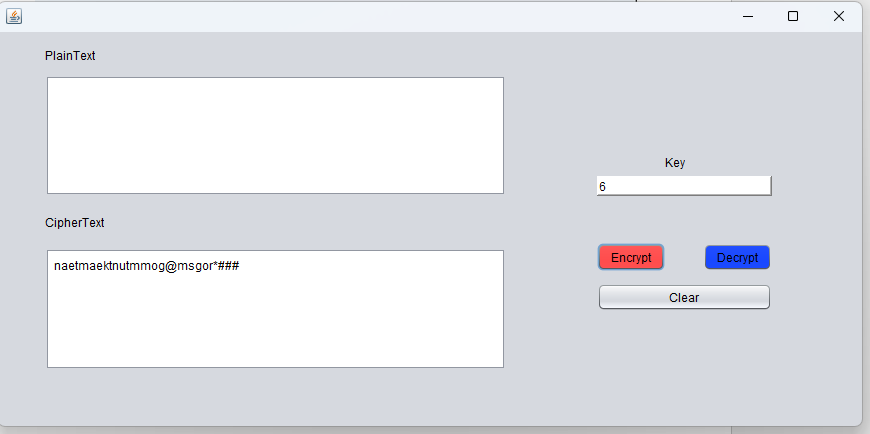


Chương trình sẽ được làm sạch.

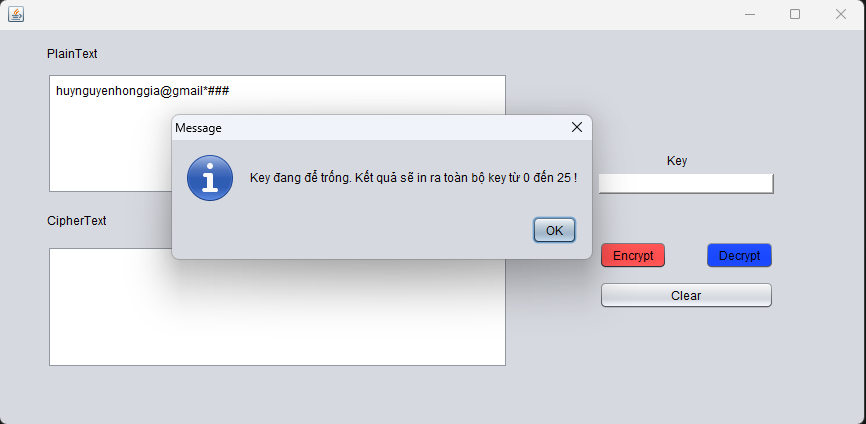
Ngoài ra ở chương trình này còn có thể sử dụng các ký hiệu đặc biệt



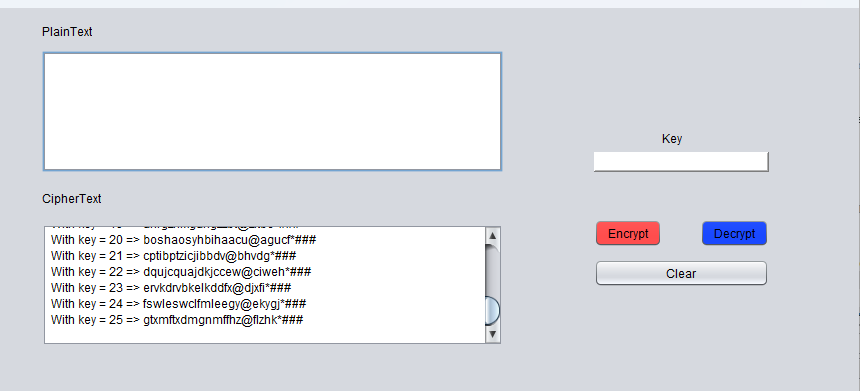
Kết quả:



Nếu không nhập key

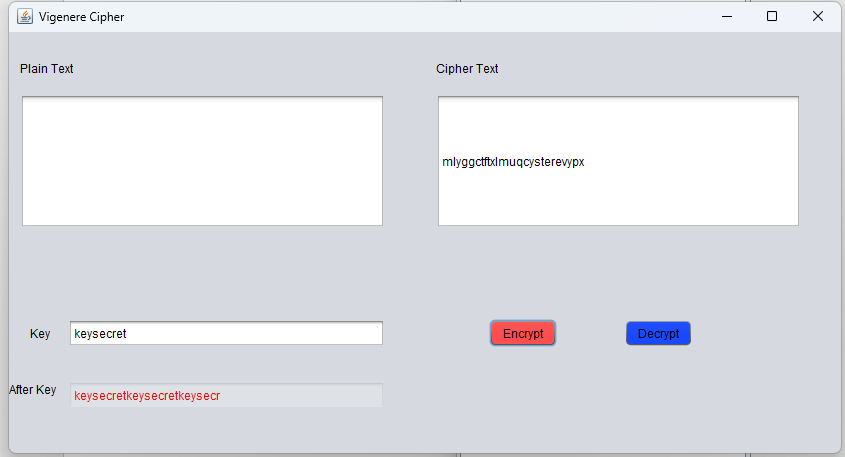


Sau đó sẽ hiển thị các key từ 0 – 25 và các đoạn văn bản được mã hóa theo từng key.

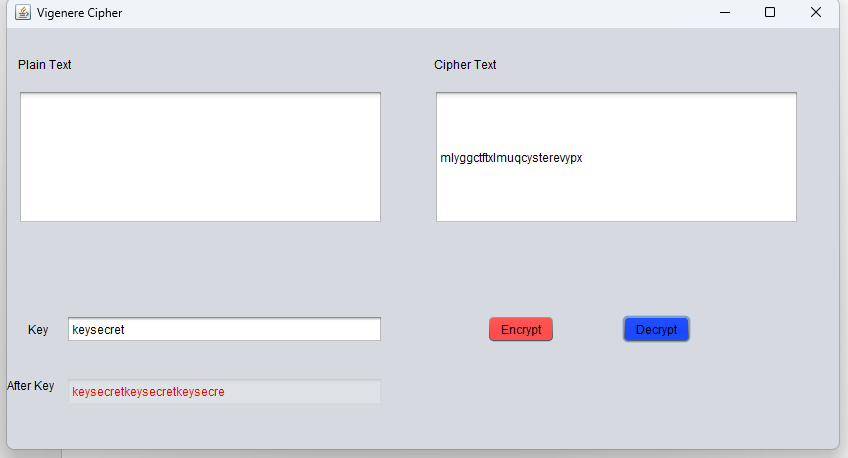


**Vegenere**

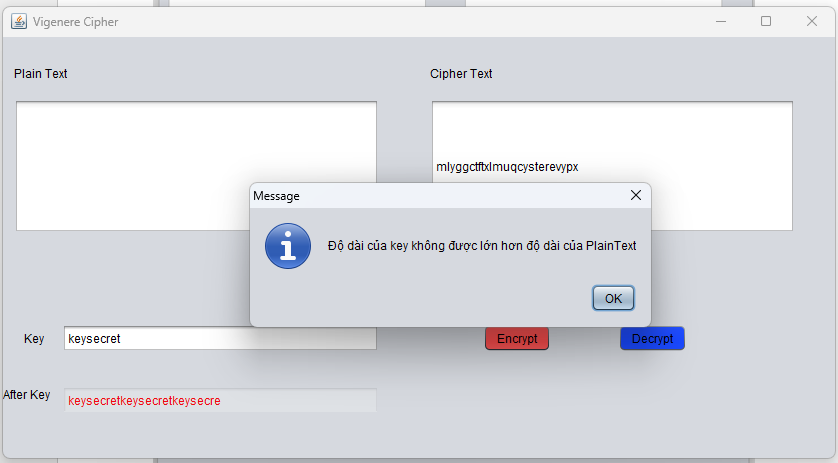
Sau khi bấm nút Encrypt ta sẽ nhận được kết quả sau :



Khi bấm nút Decrypt:

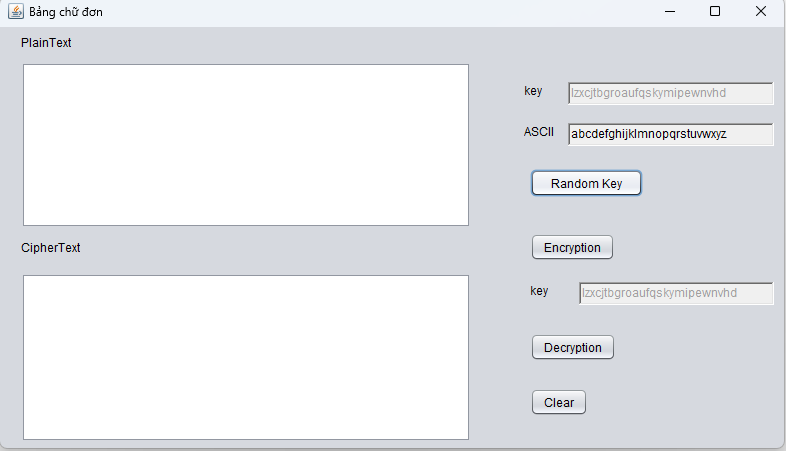


Khi ta bấm nút Encrypt 1 lần nữa:



**Bảng chữ cái đơn ( Single Alphabet)**

Khi bấm random key



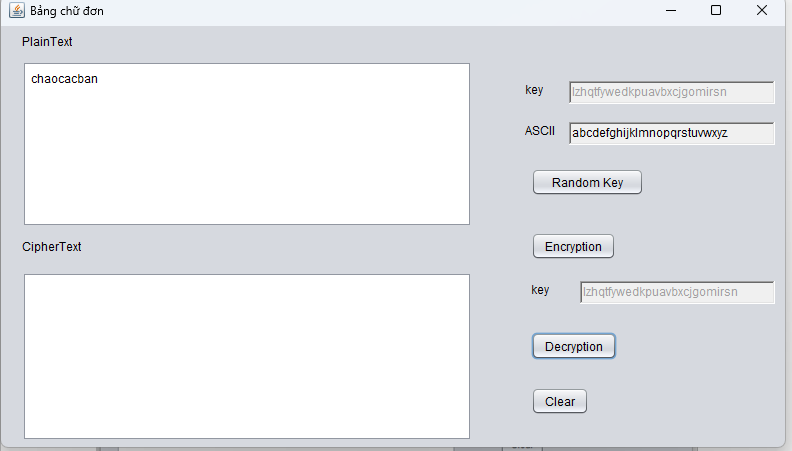
Sẽ có kết quả chuỗi key(ký tự) không trùng lặp bao gồm 26 ký tự.

Đây là khi bấm Encryption



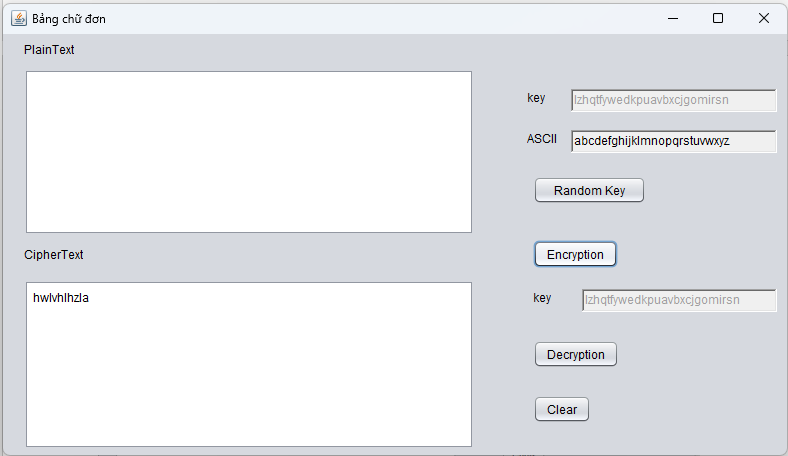
Văn bản sẽ được mã hóa

Khi bấm Decryption



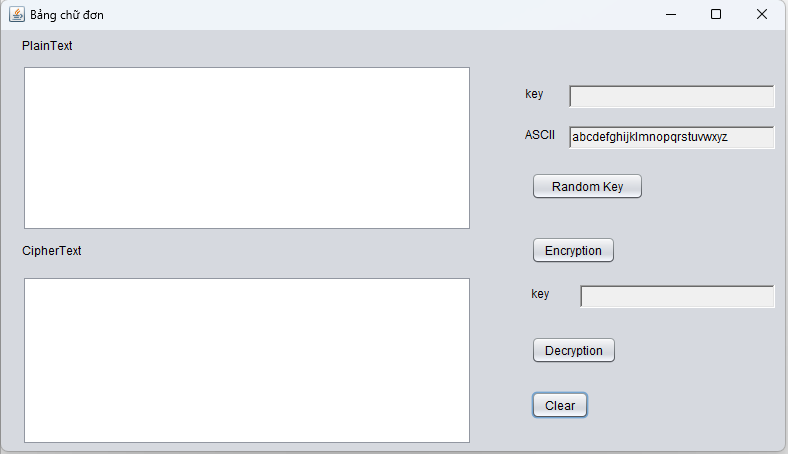
Văn bản được giải mã

Khi bấm Encryption 1 lần nữa



Văn bản vẫn mã hóa được và kết quả trả về vẫn như cũ (Tương tự đối với khi bấm Decryption )

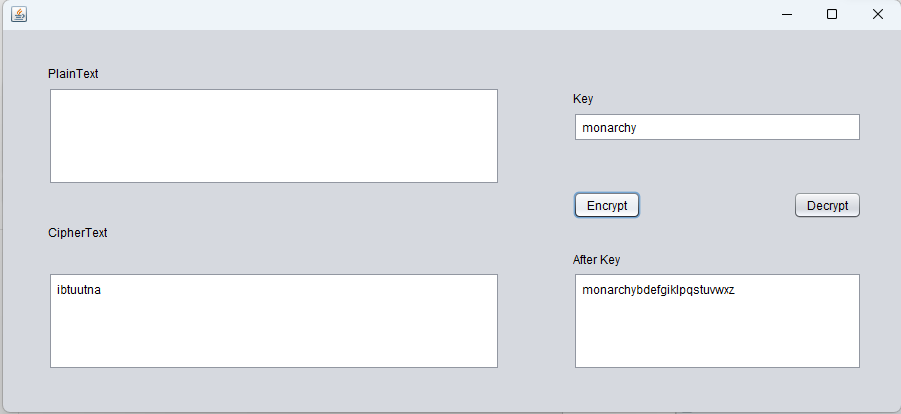
Khi bấm nút Clear



Màn hình sẽ được dọn sạch sẽ

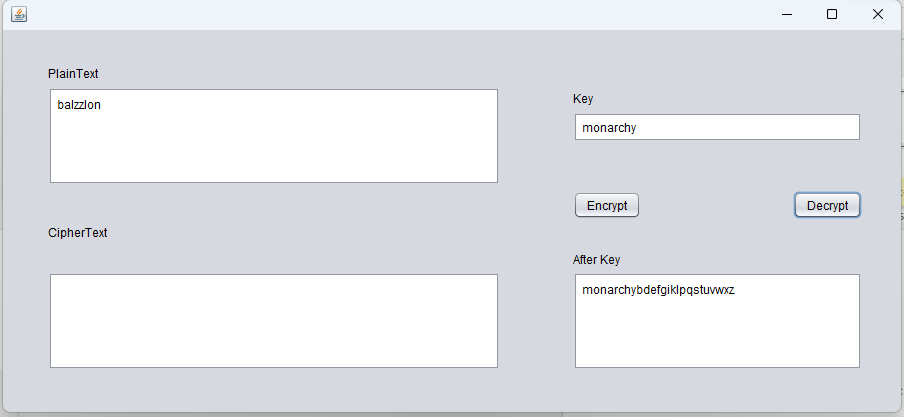
**PlayFair**

Khi bấm nút Encrypt



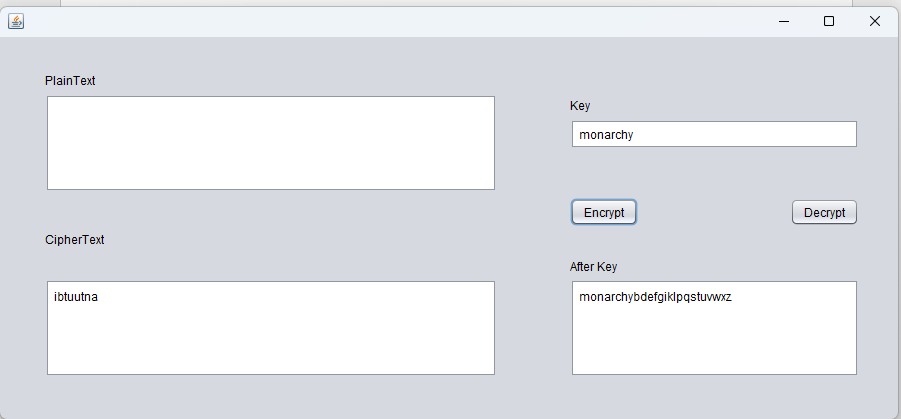
Văn bản sẽ được mã hóa

Khi bấm nút Decrypt



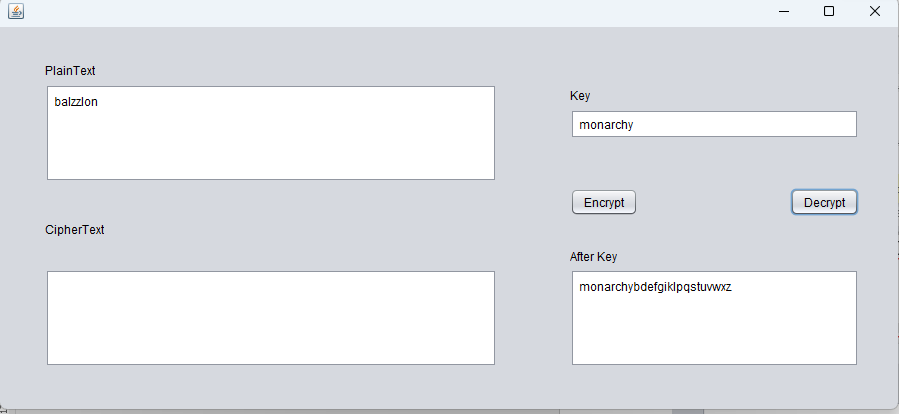
Văn bản sẽ được giải mã cùng với các ký tự dùng để mã hóa trước đó

Khi bấm Encrypt thêm 1 lần nữa



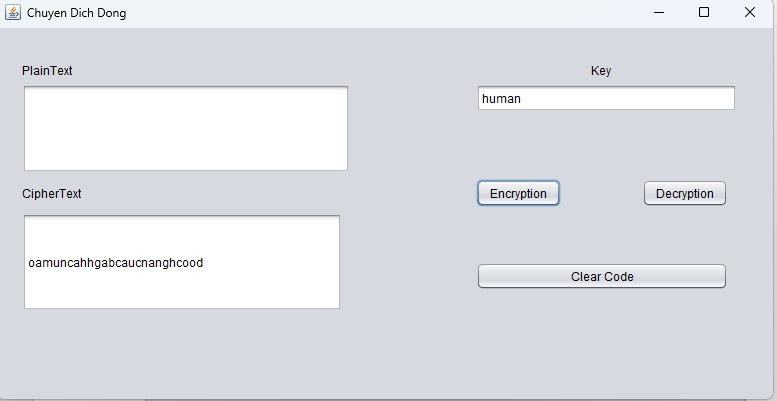
Đoạn code sẽ được mã hóa nhưng đoạn văn bản sau khi mã hóa vẫn sẽ được giữ nguyên mà không thay đổi

Tương tự vậy khi bấm Decrypt



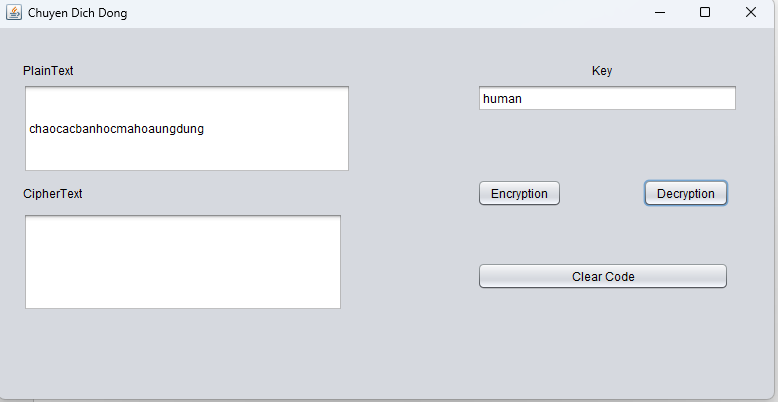
**Chuyển dịch dòng**

Khi bấm nút Encrypt



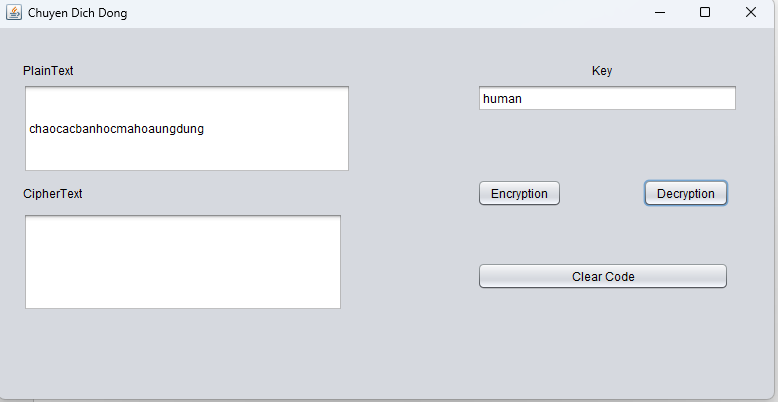
Văn bản sẽ được mã hóa

Khi bấm nút Decrypt



Văn bản được giải mã

Khi bấm nút Clear Code



Màn hình sẽ được clear sạch sẽ

CHÚNG EM CHÂN THÀNH VÀ CẢM ƠN