**Báo cáo chi tiết về ứng dụng giải mã Caesar**

**1. Giới thiệu**

* **Mục đích:** Giới thiệu về thuật toán mã hóa Caesar và mục đích của ứng dụng.
* **Các tính năng chính:** Liệt kê các tính năng chính của ứng dụng, bao gồm:
  + Mã hóa văn bản
  + Giải mã văn bản
  + Điều chỉnh khóa mã hóa
  + Giao diện người dùng trực quan

**2. Hướng dẫn sử dụng**

* **Khởi động ứng dụng:**

Chuột phải vào file ceasar.java -> chọn IDE chạy là NetBeans -> chuột phải chọn run file (shift -f6)

* **Giao diện ứng dụng:**

A screenshot of a computer

Description automatically generated

Nút giải mã

Điều chỉnh key giải mã

Nút mã hóa

Điều chỉnh key mã hóa

Text đầu ra kết quả

Text nhập đầu vào

* + **Hộp văn bản đầu vào:** Nơi người dùng nhập văn bản cần mã hóa/giải mã.
  + **Hộp văn bản đầu ra:** Hiển thị kết quả sau khi mã hóa/giải mã.
  + **Thanh trượt khóa:** Cho phép người dùng điều chỉnh khóa mã hóa.
  + **Nút mã hóa/giải mã:** Thực hiện các thao tác mã hóa và giải mã.
* **Các bước thực hiện:**
  + **Nhập văn bản:** Nhập văn bản cần mã hóa/giải mã vào hộp văn bản đầu vào.
  + **Chọn khóa:** Điều chỉnh thanh trượt khóa để chọn giá trị khóa mong muốn.
  + **Thực hiện thao tác:** Nhấn nút mã hóa hoặc giải mã để thực hiện thao tác tương ứng.
  + **Xem kết quả:** Kết quả sẽ được hiển thị tự động trong hộp văn bản đầu ra.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Mã hóa

Giải mã

**3. Giải thích thuật toán Caesar**

* **Nguyên lý hoạt động:** Cách thức hoạt động của thuật toán Caesar.

 **Chọn khóa:** Bạn chọn một số nguyên dương làm khóa. Số này sẽ quyết định số lượng vị trí mà mỗi chữ cái sẽ được dịch chuyển.

 **Mã hóa:**

* Mỗi chữ cái trong văn bản gốc sẽ được thay thế bằng chữ cái đứng sau nó một khoảng bằng giá trị của khóa.
* Ví dụ: Nếu khóa là 3, thì chữ A sẽ được thay thế bằng D, B sẽ thành E, và cứ thế.
* Khi dịch chuyển đến cuối bảng chữ cái, ta sẽ quay lại đầu bảng.

 **Giải mã:**

* Để giải mã, bạn chỉ cần dịch chuyển các chữ cái trong văn bản mã về vị trí ban đầu bằng cách trừ đi giá trị của khóa.

**4. Mã nguồn (file java)**

**5. Ưu điểm và hạn chế**

* **Ưu điểm:**
  + Giao diện đơn giản, dễ sử dụng.
  + Thuật toán dễ hiểu và triển khai.
  + Có thể tùy chỉnh khóa mã hóa.
* **Hạn chế:**
  + Độ bảo mật thấp: Dễ dàng bị bẻ khóa bằng các phương pháp tấn công đơn giản.
  + Chỉ hỗ trợ mã hóa các ký tự chữ cái.

**6. Kết luận**

* Tóm tắt lại các tính năng và khả năng của ứng dụng.
* Đề xuất các cải tiến có thể thực hiện trong tương lai (ví dụ: nâng cao độ bảo mật bằng cách kết hợp với các thuật toán khác, hỗ trợ nhiều loại mã hóa).

**Báo cáo chi tiết về ứng dụng giải mã bảng chữ đơn**

**1. Giới thiệu**

* **Mục đích**: Ứng dụng giải mã bảng chữ đơn nhằm mục đích cung cấp một công cụ đơn giản nhưng hiệu quả cho người dùng để mã hóa và giải mã văn bản bằng phương pháp thay thế, sử dụng một bảng chữ cái thay thế tùy chỉnh. Bảng chữ đơn là một trong những phương pháp mã hóa cổ điển, giúp người dùng hiểu rõ hơn về các nguyên lý mã hóa cơ bản.
* **Các tính năng chính**:
  + **Mã hóa văn bản**: Người dùng có thể nhập văn bản và sử dụng bảng chữ đơn để mã hóa văn bản đó.
  + **Giải mã văn bản**: Ứng dụng cũng hỗ trợ giải mã văn bản đã được mã hóa bằng bảng chữ đơn.
  + **Tùy chỉnh bảng chữ cái**: Người dùng có thể nhập bảng chữ cái thay thế theo ý muốn để thực hiện mã hóa/giải mã.
  + **Giao diện người dùng trực quan**: Giao diện đơn giản, dễ hiểu, giúp người dùng dễ dàng thao tác.

**2. Hướng dẫn sử dụng**

* **Khởi động ứng dụng**: Chuột phải vào file bangchudon.java -> chọn IDE chạy là NetBeans -> chuột phải chọn run file (Shift + F6).
* **Giao diện ứng dụng**:

**A screenshot of a computer

Description automatically generated**

Nơi tạo ramdom key

Nút giải mã

Nơi tạo key giải mã

Nút mã hóa

Nút ramdom

Nhập key mã hóa

Kết quả đầu ra

Nhập dãy cần mã hóa /giải mã

* + **Hộp văn bản đầu vào**: Nơi người dùng nhập văn bản cần mã hóa/giải mã.
  + **Hộp văn bản đầu ra**: Hiển thị kết quả sau khi mã hóa/giải mã.
  + **Nút mã hóa/giải mã**: Thực hiện các thao tác mã hóa và giải mã.
* **Các bước thực hiện**:
  + **Nhập văn bản**: Nhập văn bản cần mã hóa/giải mã vào hộp văn bản đầu vào.
  + **Nhập bảng chữ cái thay thế**: Nhập bảng chữ cái thay thế vào ô tương ứng.
  + **Thực hiện thao tác**: Nhấn nút mã hóa hoặc giải mã để thực hiện thao tác tương ứng.
  + **Xem kết quả**: Kết quả sẽ được hiển thị tự động trong hộp văn bản đầu ra.

**3. Giải thích thuật toán bảng chữ đơn**

* **Nguyên lý hoạt động**: Bảng chữ đơn là phương pháp mã hóa thay thế, trong đó mỗi ký tự trong văn bản gốc sẽ được thay thế bằng một ký tự khác từ một bảng chữ cái thay thế. Mỗi ký tự trong văn bản được mã hóa sẽ có một ký tự tương ứng trong bảng chữ cái thay thế.
* **Chọn bảng chữ cái thay thế**: Người dùng có thể nhập một bảng chữ cái thay thế tùy ý, miễn là nó có cùng số lượng ký tự với bảng chữ cái gốc (26 ký tự cho chữ cái Latin).
* **Mã hóa**:
  + Mỗi ký tự trong văn bản gốc sẽ được tìm kiếm trong bảng chữ cái gốc và được thay thế bằng ký tự tương ứng trong bảng chữ cái thay thế.
  + Ví dụ: Nếu bảng chữ cái thay thế là "QWERTYUIOPASDFGHJKLZXCVBNM", thì chữ A sẽ được thay thế bằng Q, B sẽ thành W, và cứ thế.
* **Giải mã**:
  + Để giải mã, bạn chỉ cần thay thế các ký tự trong văn bản mã bằng ký tự trong bảng chữ cái gốc theo bảng chữ cái thay thế đã cho.

**4. Mã nguồn (file Java)**

**5. Ưu điểm và hạn chế**

* **Ưu điểm**:
  + Giao diện thân thiện, dễ sử dụng.
  + Thuật toán dễ hiểu và triển khai.
  + Tùy chỉnh bảng chữ cái thay thế giúp linh hoạt hơn trong việc mã hóa.
* **Hạn chế**:
  + Độ bảo mật thấp: Dễ bị bẻ khóa bằng cách phân tích tần suất ký tự.
  + Chỉ hỗ trợ mã hóa các ký tự chữ cái, không hỗ trợ ký tự đặc biệt.

**6. Kết luận**

* Ứng dụng giải mã bảng chữ đơn cung cấp cho người dùng công cụ dễ dàng để mã hóa và giải mã văn bản bằng phương pháp thay thế.
* Để nâng cao độ bảo mật, có thể xem xét kết hợp với các thuật toán mã hóa khác hoặc hỗ trợ mã hóa nhiều loại ký tự hơn trong tương lai.

**Báo cáo chi tiết về ứng dụng giải mã thuật toán Playfair**

**1. Giới thiệu**

* **Mục đích**: Ứng dụng giải mã thuật toán Playfair nhằm mục đích cung cấp một công cụ mạnh mẽ và hiệu quả cho người dùng để mã hóa và giải mã văn bản thông qua phương pháp mã hóa cặp ký tự. Thuật toán Playfair là một trong những phương pháp mã hóa cổ điển, giúp người dùng nắm vững các nguyên lý mã hóa nâng cao hơn so với mã hóa thay thế đơn giản.
* **Các tính năng chính**:
  + **Mã hóa văn bản**: Người dùng có thể nhập văn bản và sử dụng ma trận Playfair để mã hóa văn bản đó.
  + **Giải mã văn bản**: Ứng dụng cũng hỗ trợ giải mã văn bản đã được mã hóa bằng thuật toán Playfair.
  + **Tùy chỉnh khóa**: Người dùng có thể nhập khóa tùy ý để tạo ma trận Playfair, từ đó thực hiện mã hóa/giải mã.
  + **Giao diện người dùng trực quan**: Giao diện đơn giản, dễ hiểu, giúp người dùng dễ dàng thao tác.

**2. Hướng dẫn sử dụng**

* **Khởi động ứng dụng**: Chuột phải vào file playfair.java -> chọn IDE chạy là NetBeans -> chuột phải chọn run file (Shift + F6).
* **Giao diện ứng dụng**:

A screenshot of a computer

Description automatically generated

Nút thực thi giải mã

Dãy còn thiếu

Nhập key giải mã

Nút thực thi mã hoá

Dãy còn thiếu

Nhập key mã hóa

Kết quả

Ô nhập plaintext

* + **Hộp văn bản đầu vào**: Nơi người dùng nhập văn bản cần mã hóa/giải mã.
  + **Hộp văn bản đầu ra**: Hiển thị kết quả sau khi mã hóa/giải mã.
  + **Nút mã hóa/giải mã**: Thực hiện các thao tác mã hóa và giải mã.
* **Các bước thực hiện**:
  + **Nhập văn bản**: Nhập văn bản cần mã hóa/giải mã vào hộp văn bản đầu vào.
  + **Nhập khóa**: Nhập khóa để tạo ma trận Playfair vào ô tương ứng.
  + **Thực hiện thao tác**: Nhấn nút mã hóa hoặc giải mã để thực hiện thao tác tương ứng.
  + **Xem kết quả**: Kết quả sẽ được hiển thị tự động trong hộp văn bản đầu ra.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Mã hóa

Giải mã

**3. Giải thích thuật toán Playfair**

* **Nguyên lý hoạt động**: Thuật toán Playfair là phương pháp mã hóa cặp ký tự, trong đó các cặp ký tự trong văn bản gốc sẽ được thay thế bằng các cặp ký tự khác dựa trên một ma trận 5x5 được tạo ra từ khóa. Mỗi ký tự trong văn bản được mã hóa sẽ được thay thế bởi một ký tự tương ứng trong ma trận.
* **Tạo ma trận Playfair**: Người dùng nhập một khóa tùy ý. Ma trận sẽ được xây dựng từ khóa đó, loại bỏ các ký tự trùng lặp và sau đó điền các ký tự còn lại từ bảng chữ cái.
* **Mã hóa**:
  + Văn bản sẽ được chia thành các cặp ký tự. Nếu có ký tự lẻ, một ký tự giả (ví dụ: 'X') sẽ được thêm vào.
  + Đối với mỗi cặp ký tự, áp dụng các quy tắc sau:
    - Nếu hai ký tự ở cùng hàng, thay thế bằng ký tự bên phải (quay vòng nếu cần).
    - Nếu hai ký tự ở cùng cột, thay thế bằng ký tự bên dưới (quay vòng nếu cần).
    - Nếu hai ký tự ở vị trí khác nhau, thay thế bằng ký tự ở giao điểm của hàng của ký tự đầu tiên và cột của ký tự thứ hai và ngược lại.
* **Giải mã**:
  + Quy trình giải mã tương tự như mã hóa, nhưng thay thế theo các quy tắc ngược lại:
    - Nếu hai ký tự ở cùng hàng, thay thế bằng ký tự bên trái.
    - Nếu hai ký tự ở cùng cột, thay thế bằng ký tự bên trên.
    - Nếu hai ký tự ở vị trí khác nhau, thay thế bằng ký tự ở giao điểm của hàng của ký tự đầu tiên và cột của ký tự thứ hai và ngược lại.

**4. Mã nguồn (file Java)**

**5. Ưu điểm và hạn chế**

* **Ưu điểm**:
  + **Giao diện thân thiện, dễ sử dụng**: Người dùng dễ dàng thao tác với giao diện trực quan.
  + **Thuật toán mạnh mẽ hơn**: Playfair cung cấp độ bảo mật cao hơn so với mã hóa thay thế đơn giản.
  + **Tùy chỉnh khóa**: Người dùng có thể tự tạo ma trận bằng cách nhập khóa riêng, tăng tính linh hoạt trong việc mã hóa.
* **Hạn chế**:
  + **Độ bảo mật chưa hoàn hảo**: Mặc dù cao hơn bảng chữ đơn, nhưng Playfair vẫn có thể bị tấn công bằng phương pháp phân tích tần suất.
  + **Chỉ hỗ trợ ký tự chữ cái**: Không hỗ trợ các ký tự đặc biệt hay khoảng trắng, cần xử lý riêng.

**6. Kết luận**

* Ứng dụng giải mã thuật toán Playfair cung cấp cho người dùng một công cụ hiệu quả để mã hóa và giải mã văn bản thông qua một phương pháp thú vị và có tính chất lịch sử.
* Để nâng cao độ bảo mật, có thể xem xét kết hợp với các thuật toán mã hóa khác hoặc hỗ trợ mã hóa nhiều loại ký tự hơn trong tương lai.

**Báo cáo chi tiết về ứng dụng giải mã Vigenère**

**1. Giới thiệu**

* **Mục đích**: Ứng dụng giải mã Vigenère nhằm cung cấp một công cụ hiệu quả cho người dùng để mã hóa và giải mã văn bản bằng phương pháp thay thế đa ký tự. Thuật toán Vigenère là một trong những phương pháp mã hóa cổ điển, giúp người dùng hiểu rõ hơn về các nguyên lý mã hóa và cách áp dụng khóa để bảo vệ thông tin.
* **Các tính năng chính**:
  + **Mã hóa văn bản**: Người dùng có thể nhập văn bản và sử dụng thuật toán Vigenère để mã hóa văn bản đó.
  + **Giải mã văn bản**: Ứng dụng cũng hỗ trợ giải mã văn bản đã được mã hóa bằng thuật toán Vigenère.
  + **Tùy chỉnh khóa**: Người dùng có thể nhập khóa tùy ý để thực hiện mã hóa/giải mã.
  + **Giao diện người dùng trực quan**: Giao diện đơn giản, dễ hiểu, giúp người dùng dễ dàng thao tác.

**2. Hướng dẫn sử dụng**

* **Khởi động ứng dụng**: Chuột phải vào file vegenereC.java -> chọn IDE chạy là NetBeans -> chuột phải chọn run file (Shift + F6).
* **Giao diện ứng dụng**:
* A screenshot of a computer

  Description automatically generated

Giải mã

Mã key tương ứng

Nhập key

Mã hóa

Key tạo tương ứng

Nhập key

Đầu ra

Đầu vào

* + **Hộp văn bản đầu vào**: Nơi người dùng nhập văn bản cần mã hóa/giải mã.
  + **Hộp văn bản đầu ra**: Hiển thị kết quả sau khi mã hóa/giải mã.
  + **Nút mã hóa/giải mã**: Thực hiện các thao tác mã hóa và giải mã.
* **Các bước thực hiện**:
  + **Nhập văn bản**: Nhập văn bản cần mã hóa/giải mã vào hộp văn bản đầu vào.
  + **Nhập khóa**: Nhập khóa vào ô tương ứng.
  + **Thực hiện thao tác**: Nhấn nút mã hóa hoặc giải mã để thực hiện thao tác tương ứng.
  + **Xem kết quả**: Kết quả sẽ được hiển thị tự động trong hộp văn bản đầu ra.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Mã hóa

Giải mã

**3. Giải thích thuật toán Vigenère**

* **Nguyên lý hoạt động**: Thuật toán Vigenère là một phương pháp mã hóa thay thế, trong đó mỗi ký tự trong văn bản gốc sẽ được thay thế bằng một ký tự khác tương ứng từ khóa. Mỗi ký tự trong văn bản được mã hóa sẽ có một ký tự tương ứng trong khóa, được lặp lại cho đến khi đủ dài.
* **Chọn khóa**: Người dùng có thể nhập một khóa tùy ý, miễn là nó chứa các ký tự chữ cái (A-Z).
* **Mã hóa**:
  + Mỗi ký tự trong văn bản gốc được tìm kiếm trong bảng chữ cái (A-Z) và được thay thế bằng ký tự trong khóa tương ứng.
  + **Ví dụ**: Nếu văn bản là "HELLO" và khóa là "KEY", thì chữ H sẽ được thay thế bằng M (H+K), E sẽ trở thành I (E+E), và cứ thế cho các ký tự còn lại.
* **Giải mã**:
  + Để giải mã, bạn chỉ cần thay thế các ký tự trong văn bản mã bằng ký tự trong khóa theo quy tắc ngược lại (trừ đi vị trí tương ứng của ký tự trong khóa).

**4. Mã nguồn (file Java)**

**5. Ưu điểm và hạn chế**

* **Ưu điểm**:
  + Giao diện thân thiện, dễ sử dụng.
  + Thuật toán bảo mật cao hơn so với bảng chữ đơn, nhờ vào việc sử dụng khóa.
  + Tùy chỉnh khóa giúp người dùng linh hoạt trong việc mã hóa.
* **Hạn chế**:
  + Độ bảo mật vẫn có thể bị giảm nếu khóa quá ngắn hoặc có tính lặp lại cao.
  + Chỉ hỗ trợ mã hóa các ký tự chữ cái, không hỗ trợ ký tự đặc biệt và số.

**6. Kết luận**

* Ứng dụng giải mã Vigenère cung cấp cho người dùng công cụ dễ dàng để mã hóa và giải mã văn bản bằng phương pháp thay thế.
* Để nâng cao độ bảo mật, có thể xem xét kết hợp với các thuật toán mã hóa khác hoặc hỗ trợ mã hóa nhiều loại ký tự hơn trong tương lai.

**Báo cáo chi tiết về ứng dụng Columnar Transposition Cipher (chuyển dịch dòng)**

**1. Giới thiệu**

* **Mục đích**: Ứng dụng này được phát triển để mã hóa và giải mã văn bản sử dụng thuật toán **Columnar Transposition Cipher**. Mục tiêu chính của ứng dụng là cung cấp một phương thức đơn giản nhưng hiệu quả để bảo mật thông tin thông qua việc sắp xếp lại các ký tự trong văn bản.
* **Các tính năng chính**:
  + **Mã hóa văn bản**: Cho phép người dùng nhập văn bản và khóa để mã hóa thông tin.
  + **Giải mã văn bản**: Hỗ trợ giải mã văn bản đã được mã hóa bằng cách sử dụng khóa tương ứng.
  + **Điều chỉnh khóa mã hóa**: Người dùng có thể nhập và điều chỉnh khóa để thay đổi cách thức mã hóa.
  + **Giao diện người dùng trực quan**: Ứng dụng cung cấp một giao diện thân thiện, dễ sử dụng cho người dùng.

**2. Hướng dẫn sử dụng**

* **Khởi động ứng dụng**:
  1. Mở file chuyendichdong.java.
  2. Chọn IDE NetBeans để chạy ứng dụng.
  3. Nhấp chuột phải vào file và chọn "Run File" (hoặc sử dụng phím tắt Shift + F6).
* **Giao diện ứng dụng**:

A screenshot of a computer

Description automatically generated

Giải mã

Nhập key

Mã hóa

Nhập key

Kết quả

Nhập plaintext

* 1. **Hộp văn bản đầu vào**: Nơi người dùng nhập văn bản cần mã hóa hoặc giải mã.
  2. **Hộp văn bản đầu ra**: Hiển thị kết quả sau khi thực hiện mã hóa hoặc giải mã.
  3. **Hộp văn bản nhập khóa**: Người dùng nhập khóa cần thiết để mã hóa hoặc giải mã.
  4. **Nút mã hóa/giải mã**: Thực hiện các thao tác mã hóa và giải mã.
* **Các bước thực hiện**:
  1. **Nhập văn bản**: Nhập văn bản cần mã hóa hoặc giải mã vào hộp văn bản đầu vào.
  2. **Nhập khóa**: Nhập khóa vào hộp văn bản tương ứng.
  3. **Thực hiện thao tác**: Nhấn nút mã hóa hoặc giải mã để thực hiện thao tác tương ứng.
  4. **Xem kết quả**: Kết quả sẽ được hiển thị tự động trong hộp văn bản đầu ra.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Mã hóa

Giải mã

**3. Giải thích thuật toán Columnar Transposition Cipher**

* **Nguyên lý hoạt động**:
  + Thuật toán này sử dụng một bảng để sắp xếp các ký tự của văn bản theo hàng và cột. Sau đó, thứ tự cột được thay đổi dựa trên một khóa đã định.
* **Chọn khóa**:
  + Khóa là một chuỗi ký tự mà từ đó các cột trong bảng sẽ được sắp xếp lại. Người dùng nhập khóa vào ứng dụng, và độ dài của khóa quyết định số cột trong bảng.
* **Mã hóa**:
  + Văn bản gốc được điền vào bảng theo hàng. Sau khi điền đầy đủ, các cột được sắp xếp lại theo thứ tự được xác định bởi khóa. Kết quả mã hóa là việc đọc các cột theo thứ tự mới.
  + **Ví dụ**:
    - Giả sử khóa là "HACK", và văn bản cần mã hóa là "HELLO WORLD". Bảng có thể được điền như sau:

mathematica

Copy code

H A C K

H E L O

W O R L

D

* + - Sau đó, cột sẽ được sắp xếp theo thứ tự của khóa. Kết quả mã hóa sẽ là các ký tự được đọc từ các cột theo thứ tự đã được sắp xếp.
* **Giải mã**:
  + Để giải mã, bạn cần xác định lại thứ tự các cột và đọc các ký tự theo hàng để khôi phục văn bản gốc.

**4. Mã nguồn (file java)**

**5. Ưu điểm và hạn chế**

* **Ưu điểm**:
  + Giao diện đơn giản, dễ sử dụng.
  + Thuật toán dễ hiểu và triển khai.
  + Có thể tùy chỉnh khóa mã hóa.
* **Hạn chế**:
  + Độ bảo mật thấp: Dễ dàng bị bẻ khóa bằng các phương pháp tấn công đơn giản.
  + Chỉ hỗ trợ mã hóa các ký tự chữ cái.

**6. Kết luận**

* **Tóm tắt**: Ứng dụng giải mã **Columnar Transposition Cipher** cho phép người dùng mã hóa và giải mã văn bản một cách dễ dàng. Nó hỗ trợ điều chỉnh khóa và cung cấp giao diện trực quan.
* **Đề xuất cải tiến**: Có thể nâng cao độ bảo mật bằng cách kết hợp với các thuật toán mã hóa khác, hỗ trợ mã hóa nhiều loại ký tự và cải thiện giao diện người dùng để tăng tính thân thiện.