LAB 3 – MÃ HÓA ĐỐI XỨNG CỔ ĐIỂN, MÃ DÒNG, MÃ KHỐI VÀ MÃ FEISTEL

Author: Trần Quý Nam

Date: 24/3/2025

Sinh viên cài đặt thực thi các chương trình sau dùng C++:

Bài 1. Mã hóa thay thế. Cài đặt chương trình C++ thực hiện mã hóa thay thế có các menu thực hiện các chức năng:

- Chọn mã hóa thay thế cho tiếng Việt và tiếng Anh
- Nhập khóa K bất kỳ
- Chọn mã hóa hoặc giải mã với xâu bất kỳ nhập vào từ bàn phím

Bài 2. Mã hóa hoán vị. Viết chương trình C++ để mã hóa một chuỗi bằng phương pháp hoán vị với một khóa đã cho.

- Nhập một chuỗi văn bản không dấu.
- Nhập một hoán vị của chuỗi (khóa).
- Xuất chuỗi đã được mã hóa.

Bài 3. Mã hóa hoán vị theo cột (Columnar Transposition Cipher)

Viết chương trình C++ để mã hóa một chuỗi bằng phương pháp hoán vị theo cột với một từ khóa.

- Chuyển chuỗi thành ma trận theo số cột là độ dài từ khóa.
- Sắp xếp các cột theo thứ tự từ khóa đã sắp xếp.

- Đọc các cột theo thứ tự để tạo ra bản mã.

Bài 4. Mã hóa dòng

Viết chương trình C++ để mã hóa và giải mã một chuỗi văn bản bằng phép toán XOR với một khóa đơn ký tự.

- Nhập một chuỗi văn bản.
- Nhập một ký tự làm khóa.
- Áp dụng phép XOR giữa từng ký tự của văn bản với khóa.

Gợi ý: Dùng phép XOR ^ trong C++.

Bài 5. Mã hóa khối

Viết chương trình C++ để mã hóa một chuỗi bằng cách chia thành các khối có kích thước cố định và thực hiện phép XOR với một khóa.

- Nhập một chuỗi văn bản.
- Nhập một khóa có độ dài bằng với kích thước khối.
- Chia văn bản thành các khối có kích thước n.
- Mã hóa mỗi khối bằng phép XOR với khóa.

Ví dụ:

Plaintext: HELLO WORLD

Block Size: 3

Key: XYZ

Gợi ý: Dùng phép XOR ^ để mã hóa từng ký tự trong từng khối.

Bài 6. Cài đặt thuật toán Feistel

Viết chương trình C++ để cài đặt thuật toán mã hóa khối Feistel theo các bước sau:

- Nhập một khối dữ liệu dạng chuỗi nhị phân.
- Chia thành hai nửa: L (left) và R (right).
- Thực hiện n vòng Feistel, mỗi vòng có:

  $L_{i+1} = R_i$

  $R_{i+1} = L_i \oplus F(R_i, K_i)$

- Hàm F là một phép XOR
- Khóa vòng (subkey) được sinh từ một khóa chính
- Hiển thị kết quả của từng vòng.
- Giải mã dữ liệu bằng cách đảo ngược các bước.

Ví dụ:

Plaintext: 10100111

Key: 1101

Rounds: 3

Khi đó:

Mã hóa là:

Round 1: L = 1010, R = 0111

Round 2: L = 0111, R = 1100

Round 3: L = 1100, R = 1011

Ciphertext: 11001011

Giải mã là:

Round 1: L = 1011, R = 1100

Round 2: L = 1100, R = 0111

Round 3: L = 0111, R = 1010

Decrypted Text: 10100111
