Tugas 3
Kriptografi

Disusun Oleh:

Rayhan Nugrah Krisitio – 140810220075

PROGRAM STUDI S-1 TEKNIK INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM UNIVERSITAS PADJADJARAN
JATINANGOR
2024

1. Buatlah program untuk enkripsi, dekripsi, dan mencari kunci Hill Cipher (bahasa pemrograman bebas)

   github : https://github.com/rayhanuug/75-Kripto24.git

Code :

```
#include <iostream>
#include <string>
using namespace std;

// Fungsi untuk mengalikan matriks dengan vektor
void multiplyMatrix(int key[2][2], int message[2], int result[2]) {
    for(int i = 0; i < 2; i++) {
        result[i] = 0;
        for(int j = 0; j < 2; j++) {
            result[i] += key[i][j] * message[j];
        }
        result[i] = result[i] % 26;
    }
}

// Fungsi untuk mencari determinan matriks 2x2
int determinant(int key[2][2]) {
    return (key[0][0] * key[1][1] - key[0][1] * key[1][0]);
}

// Fungsi untuk mencari modular multiplicative inverse
int modInverse(int a) {
    a = a % 26;
```

```
    for(int x = 1; x < 26; x++) {
        if((a * x) % 26 == 1) {
            return x;
        }
    }
    return 1;
}

// Fungsi enkripsi
string encrypt(string message, int key[2][2]) {
    string result = "";
    int i = 0;

    // Proses setiap pasangan karakter
    while(i < message.length()) {
        int messageVector[2] = {
            message[i] - 'A',
            (i + 1 < message.length()) ? message[i + 1] - 'A' : 0
        };

        int resultVector[2];
        multiplyMatrix(key, messageVector, resultVector);

        result += (char)(resultVector[0] + 'A');
        result += (char)(resultVector[1] + 'A');

        i += 2;
    }
```

```
    return result;
}

// Fungsi dekripsi
string decrypt(string cipher, int key[2][2]) {
    // Hitung determinan dan inverse
    int det = determinant(key);
    det = ((det % 26) + 26) % 26;
    int detInv = modInverse(det);

    // Hitung matriks adjoin
    int adj[2][2] = {
        {key[1][1], -key[0][1]},
        {-key[1][0], key[0][0]}
    };

    // Hitung inverse key
    int inverse[2][2];
    for(int i = 0; i < 2; i++) {
        for(int j = 0; j < 2; j++) {
            inverse[i][j] = (adj[i][j] * detInv) % 26;
            if(inverse[i][j] < 0) inverse[i][j] += 26;
        }
    }

    // Gunakan inverse key untuk dekripsi
    return encrypt(cipher, inverse);
}
```

```
// Fungsi untuk mencari kunci
void findKey(string plain, string cipher) {
    int P[2][2], C[2][2];

    // Ubah plain text dan cipher text ke bentuk matriks
    for(int i = 0; i < 2; i++) {
        P[i][0] = plain[i] - 'A';
        P[i][1] = plain[i+2] - 'A';
        C[i][0] = cipher[i] - 'A';
        C[i][1] = cipher[i+2] - 'A';
    }

    // Hitung determinan P
    int det = determinant(P);
    det = ((det % 26) + 26) % 26;
    int detInv = modInverse(det);

    // Hitung adjoin P
    int adj[2][2] = {
        {P[1][1], -P[0][1]},
        {-P[1][0], P[0][0]}
    };

    // Hitung inverse P
    int PInv[2][2];
    for(int i = 0; i < 2; i++) {
        for(int j = 0; j < 2; j++) {
            PInv[i][j] = (adj[i][j] * detInv) % 26;
            if(PInv[i][j] < 0) PInv[i][j] += 26;
```

```
        }
    }

    // Hitung key = C × P^(-1)
    int key[2][2];
    for(int i = 0; i < 2; i++) {
        for(int j = 0; j < 2; j++) {
            key[i][j] = 0;
            for(int k = 0; k < 2; k++) {
                key[i][j] += C[i][k] * PInv[k][j];
            }
            key[i][j] = key[i][j] % 26;
            if(key[i][j] < 0) key[i][j] += 26;
        }
    }

    // Tampilkan hasil
    cout << "Matriks kunci yang ditemukan:\n";
    for(int i = 0; i < 2; i++) {
        for(int j = 0; j < 2; j++) {
            cout << key[i][j] << " ";
        }
        cout << endl;
    }
}

int main() {
    int choice;
    cout << "Hill Cipher Menu:\n";
```

```
    cout << "1. Enkripsi\n";
    cout << "2. Dekripsi\n";
    cout << "3. Cari Kunci\n";
    cout << "Pilihan: ";
    cin >> choice;

    if(choice == 1 || choice == 2) {
        int key[2][2];
        cout << "Masukkan matriks kunci 2x2:\n";
        for(int i = 0; i < 2; i++) {
            for(int j = 0; j < 2; j++) {
                cin >> key[i][j];
            }
        }

        string text;
        cout << "Masukkan teks (KAPITAL): ";
        cin >> text;

        if(choice == 1) {
            cout << "Hasil enkripsi: " << encrypt(text, key) << endl;
        } else {
            cout << "Hasil dekripsi: " << decrypt(text, key) << endl;
        }
    }
    else if(choice == 3) {
        string plaintext, ciphertext;
        cout << "Masukkan plaintext (4 huruf KAPITAL): ";
        cin >> plaintext;
```

```
        cout << "Masukkan ciphertext (4 huruf KAPITAL): ";
        cin >> ciphertext;

        findKey(plaintext, ciphertext);
    }

    return 0;
}
```

Running Code :

```
PS C:\Users\raiha\kodingan\semester ganjil\semester 5\Praktikum\Kriptografi\Tugas3> ./p
Hill Cipher Menu:
1. Enkripsi
2. Dekripsi
3. Cari Kunci
Pilihan: 2
Masukkan matriks kunci 2x2:
3
2
2
7
Masukkan teks (KAPITAL): MJCRHG
Hasil dekripsi: KRIPTO
PS C:\Users\raiha\kodingan\semester ganjil\semester 5\Praktikum\Kriptografi\Tugas3> ./p
Hill Cipher Menu:
1. Enkripsi
2. Dekripsi
3. Cari Kunci
Pilihan: 3
Masukkan plaintext (4 huruf KAPITAL): ICON
Masukkan ciphertext (4 huruf KAPITAL): AKDO
Matriks kunci yang ditemukan:
20 24
24 24
PS C:\Users\raiha\kodingan\semester ganjil\semester 5\Praktikum\Kriptografi\Tugas3>
```