CSA5122-CRYPTOGRAPHY FOR NETWORK AND SECURITY

LAB PROGRAMS EXECUTION

3.HILL CIPHER

#include <stdio.h>

#include <string.h>

#include <ctype.h>

#define SIZE 2

void multiply(int key[SIZE][SIZE], int vec[SIZE], int res[SIZE]) {

for (int i = 0; i < SIZE; i++) {

res[i] = 0;

for (int j = 0; j < SIZE; j++)

res[i] += key[i][j] \* vec[j];

res[i] %= 26;

}

}

int modInv(int a) {

for (int x = 1; x < 26; x++)

if ((a \* x) % 26 == 1) return x;

return -1;

}

int inverse(int key[SIZE][SIZE], int inv[SIZE][SIZE]) {

int det = (key[0][0]\*key[1][1] - key[0][1]\*key[1][0] + 26) % 26;

int invDet = modInv(det);

if (invDet == -1) return 0;

inv[0][0] = key[1][1] \* invDet % 26;

inv[0][1] = -key[0][1] \* invDet % 26;

inv[1][0] = -key[1][0] \* invDet % 26;

inv[1][1] = key[0][0] \* invDet % 26;

for (int i = 0; i < SIZE; i++)

for (int j = 0; j < SIZE; j++)

inv[i][j] = (inv[i][j] + 26) % 26;

return 1;

}

void process(char \*text, int key[SIZE][SIZE]) {

int vec[SIZE], res[SIZE];

for (int i = 0; i < strlen(text); i += 2) {

vec[0] = toupper(text[i]) - 'A';

vec[1] = toupper(text[i+1]) - 'A';

multiply(key, vec, res);

printf("%c%c", res[0] + 'A', res[1] + 'A');

}

printf("\n");

}

int main() {

int key[SIZE][SIZE] = {{3, 3}, {2, 5}}, inv[SIZE][SIZE];

char input[100];

int ch;

printf("Hill Cipher\n1. Encrypt\n2. Decrypt\nEnter choice: ");

scanf("%d", &ch); getchar();

printf("Enter text (A-Z, even length): ");

fgets(input, sizeof(input), stdin);

input[strcspn(input, "\n")] = 0;

if (strlen(input) % 2) strcat(input, "X");

if (ch == 1)

printf("Encrypted: "), process(input, key);

else if (ch == 2 && inverse(key, inv))

printf("Decrypted: "), process(input, inv);

else

printf("Invalid choice or key not invertible.\n");

return 0;

}

