10. Write a C program for Playfair matrix:

M F H I/J K

U N O P Q

Z V W X Y

E L A R G

D S T B C

Encrypt this message: Must see you over Cadogan West. Coming at once.

#include <stdio.h>

#include <string.h>

#include <ctype.h>

#define SIZE 5

char matrix[SIZE][SIZE] = {

{'M', 'F', 'H', 'I', 'K'},

{'U', 'N', 'O', 'P', 'Q'},

{'Z', 'V', 'W', 'X', 'Y'},

{'E', 'L', 'A', 'R', 'G'},

{'D', 'S', 'T', 'B', 'C'}

};

void findPosition(char ch, int \*row, int \*col) {

if (ch == 'J') ch = 'I';

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

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

if (matrix[i][j] == ch) {

\*row = i;

\*col = j;

return;

}

}

}

}

void encryptPair(char a, char b, char \*res) {

int row1, col1, row2, col2;

findPosition(a, &row1, &col1);

findPosition(b, &row2, &col2);

if (row1 == row2) {

res[0] = matrix[row1][(col1 + 1) % SIZE];

res[1] = matrix[row2][(col2 + 1) % SIZE];

} else if (col1 == col2) {

res[0] = matrix[(row1 + 1) % SIZE][col1];

res[1] = matrix[(row2 + 1) % SIZE][col2];

} else {

res[0] = matrix[row1][col2];

res[1] = matrix[row2][col1];

}

}

void prepareText(char text[], char pairs[][2], int \*pairCount) {

char clean[500];

int len = 0;

for (int i = 0; text[i]; i++) {

if (isalpha(text[i])) {

clean[len++] = toupper(text[i] == 'J' ? 'I' : text[i]);

}

}

int i = 0;

\*pairCount = 0;

while (i < len) {

char a = clean[i];

char b = (i + 1 < len) ? clean[i + 1] : 'X';

if (a == b) {

b = 'X';

i++;

} else {

i += 2;

}

pairs[\*pairCount][0] = a;

pairs[\*pairCount][1] = b;

(\*pairCount)++;

}

}

void encryptMessage(char text[]) {

char pairs[250][2];

int count;

char res[3];

res[2] = '\0';

prepareText(text, pairs, &count);

printf("Encrypted Message:\n");

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

encryptPair(pairs[i][0], pairs[i][1], res);

printf("%s", res);

}

printf("\n");

}

int main() {

char message[] = "Must see you over Cadogan West. Coming at once.";

printf("Original Message:\n%s\n\n", message);

encryptMessage(message);

return 0;

}

