

```

#include "stdafx.h"
#include <iostream>

using namespace std;

char letters[] = {'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p',
                 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z'};

size_t letters_size = 26;

size_t index_of_letter(char s) {
    for (size_t i = 0; i < letters_size; ++i) {
        if (letters[i] == s) {
            return i;
        }
    }
    return -1;
}

char encrypt_letter(char s, size_t key) {
    size_t s_index = index_of_letter(s);
    size_t res_index = (s_index + key) % letters_size;
    return letters[res_index];
}

char* encrypt(int key, char* message, size_t message_length) {
    char* result = new char[message_length];
    for (size_t i = 0; i < message_length; ++i) {
        result[i] = encrypt_letter(message[i], key);
    }
    return result;
}

char decrypt_letter(char s, size_t key) {
    size_t s_index = index_of_letter(s);

```

```

size_t res_index = s_index - key;
if (res_index < 0) {
    res_index = res_index + letters_size;
}
return letters[res_index];
}

```

```

char* decrypt(int key, char* message, size_t message_length) {
    char* result = new char[message_length];
    for (size_t i = 0; i < message_length; ++i) {
        result[i] = decrypt_letter(message[i], key);
    }
    return result;
}

```

```

void show_message(char* message, size_t message_length) {
    for (size_t i = 0; i < message_length; ++i) {
        cout<<message[i];
    }
    cout<<endl;
}

```

```

int main()
{
    char* message = new char[1000];
    size_t message_length = 0;
    int key = 10;

    cout<<"Enter message letter one by one and finishing by the symbol ; "<<endl;
    char curr = ' ';
    size_t i = 0;
    while (curr != ';') {
        cin>>curr;
        if (curr != ';') {

```

```
        message[i] = curr;
        ++i;
    }
}
message_length = i;

cout<<"Input message: ";
show_message(message, message_length);

char* encrypted_message = encrypt(key, message, message_length);
cout<<"Encrypted message: ";
show_message(encrypted_message, message_length);

char* decrypted_message = decrypt(key, encrypted_message, message_length);
cout<<"Decrypted message: ";
show_message(decrypted_message, message_length);
return 0;
}
```