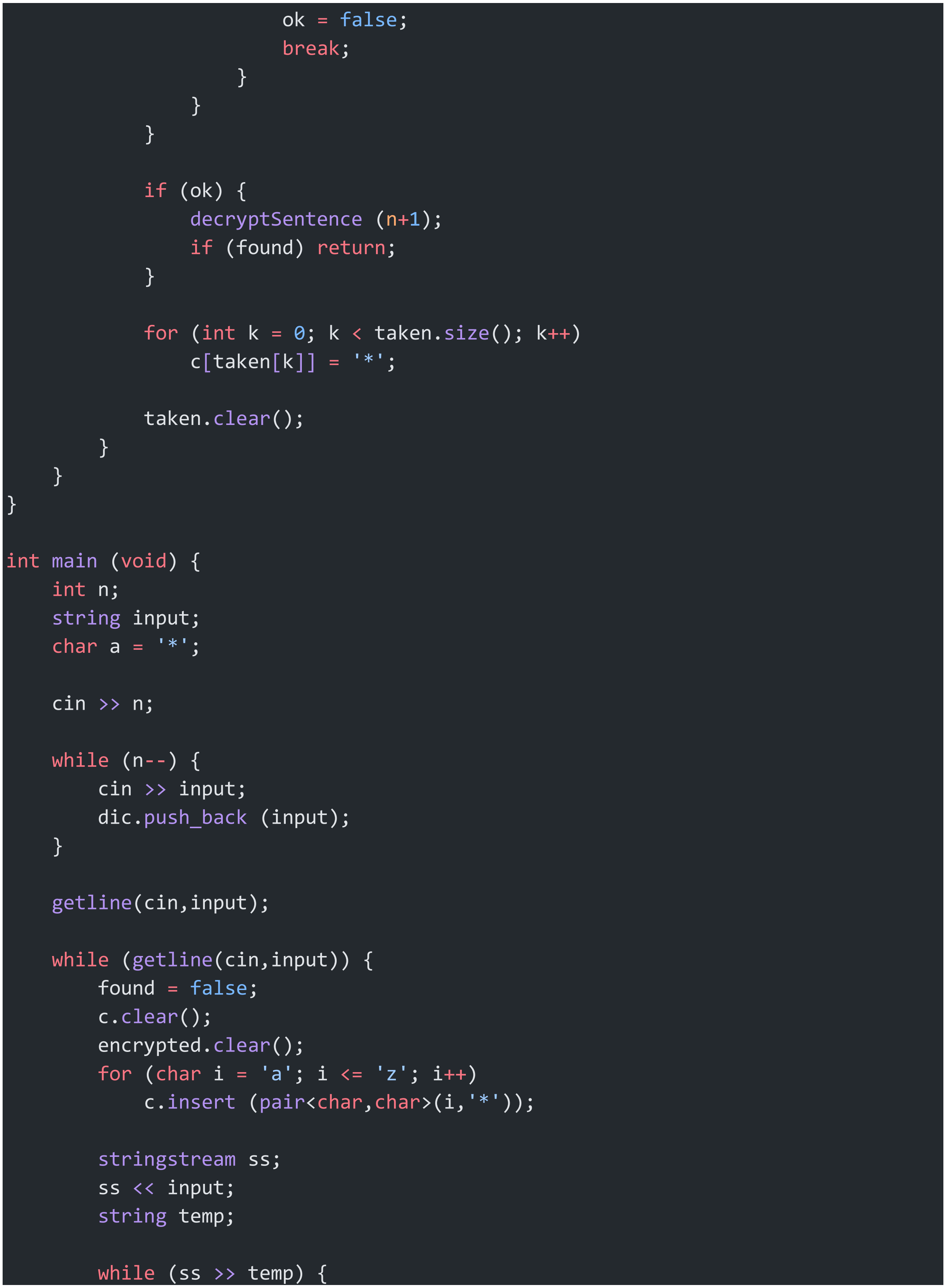
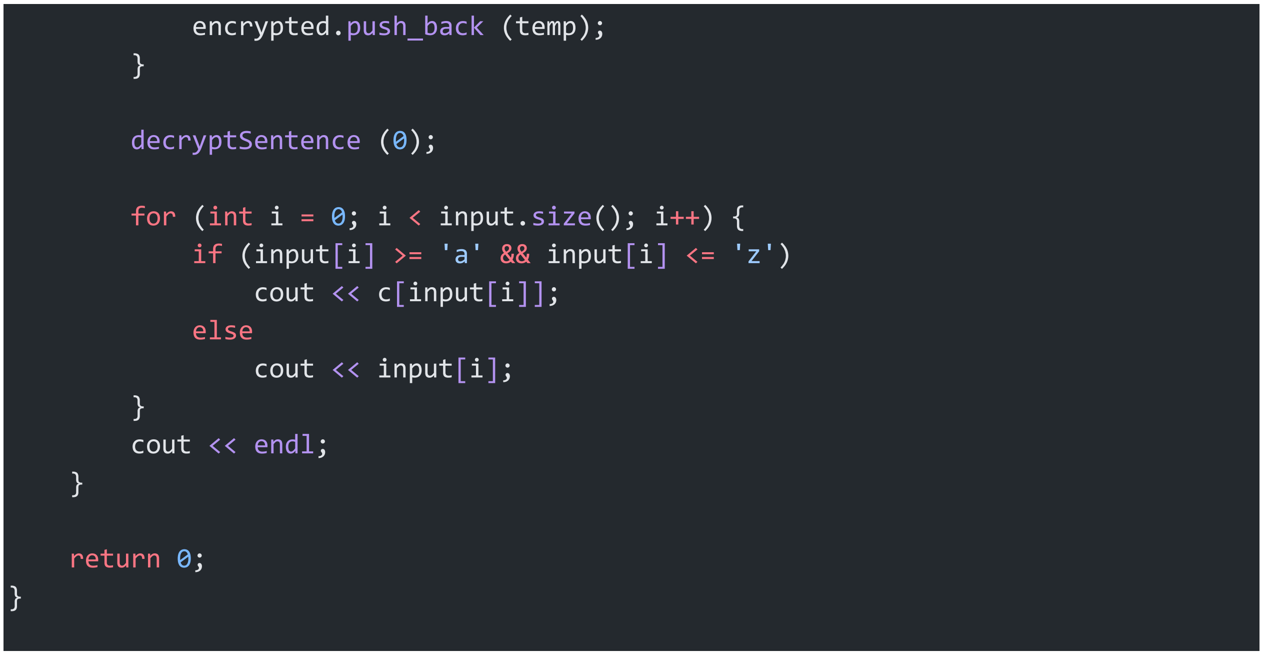
Crypt kicker

Program:

```
# #include <iostream>

#include <string>
#include <vector>
#include <map>
#include <sstream>
using namespace std;

vector<string> dic;
vector<string> encrypted;
map<char,char> c;
bool found;

void decryptSentence (int n) {
    if (found) return;
    if (n == encrypted.size()) {
        found = true;
        return;
    }

    vector<char> taken;

    for (int i = 0; i < dic.size(); i++) {
        if (dic[i].size() == encrypted[n].size()) {
            bool ok = true;
            for (int j = 0; j < dic[i].size(); j++) {
                if (c[encrypted[n][j]] == '*') {
                    for (map<char,char>::iterator it = c.begin(); it != c.end();
it++) {
                        if ((*it).second == dic[i][j]) {
                            ok = false;
                            break;
                        }
                    }
                    if (!ok) break;

                    taken.push_back (encrypted[n][j]);
                    c[encrypted[n][j]] = dic[i][j];
                } else {
                    if (c[encrypted[n][j]] != dic[i][j]) {
```

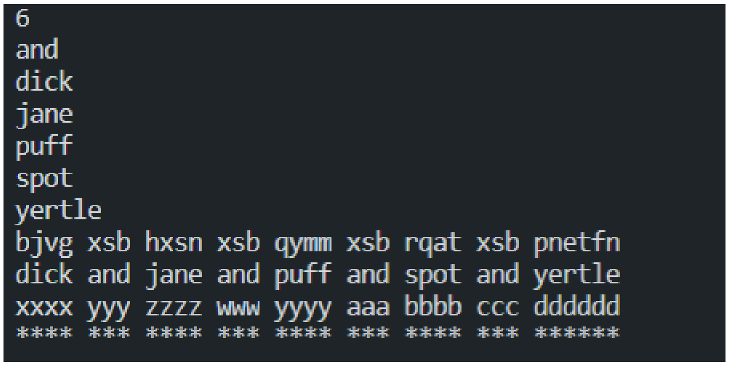


```
                        ok = false;
                        break;
                    }
                }
            }

            if (ok) {
                decryptSentence (n+1);
                if (found) return;
            }

            for (int k = 0; k < taken.size(); k++)
                c[taken[k]] = '*';

            taken.clear();
        }
    }
}

int main (void) {
    int n;
    string input;
    char a = '*';

    cin >> n;

    while (n--) {
        cin >> input;
        dic.push_back (input);
    }

    getline(cin,input);

    while (getline(cin,input)) {
        found = false;
        c.clear();
        encrypted.clear();
        for (char i = 'a'; i <= 'z'; i++)
            c.insert (pair<char,char>(i,'*'));

        stringstream ss;
        ss << input;
        string temp;

        while (ss >> temp) {
```



```
            encrypted.push_back (temp);
        }

        decryptSentence (0);

        for (int i = 0; i < input.size(); i++) {
            if (input[i] >= 'a' && input[i] <= 'z')
                cout << c[input[i]];
            else
                cout << input[i];
        }
        cout << endl;
    }

    return 0;
}
```

Output:



```
6
and
dick
jane
puff
spot
yertle
bjvg xsb hxsn xsb qymm xsb rqat xsb pnetfn
dick and jane and puff and spot and yertle
xxxx yyy zzzz www yyyy aaa bbbb ccc dddddd
**** *** **** *** **** *** **** *** ******
```