A

#ifdef \_MSC\_VER

#define \_CRT\_SECURE\_NO\_WARNINGS

#endif

#include <iostream>

#include <string>

#include <vector>

#include <algorithm>

using namespace std;

int main() {

    freopen("huffman.in", "r", stdin);

    freopen("huffman.out", "w", stdout);

    int n;

    unsigned long long ans = 0, p;

    scanf("%i", &n);

    vector <unsigned long long int> a;

    vector <unsigned long long int> b;

    a.reserve(n);

    b.reserve(n);

    for (int i = 0; i < n; i++) {

        scanf("%lld", &p);

        a.push\_back(p);

        b.push\_back(LLONG\_MAX);

    }

    sort(a.begin(), a.end());

    int i = 0, j = 0;

    for (int k = 0; k < n; k++) {

        if (i + 1 < n && (j >= n || a.at(i) + a.at(i + 1) <= a.at(i) + b.at(j)) &&

            (j + 1 >= n || a.at(i) + a.at(i + 1) <= b.at(j) + b.at(j + 1))) {

            b.at(k) = a.at(i) + a.at(i + 1);

            ans += b.at(k);

            i += 2;

            continue;

        }

        if ((i < n && j < n) && (i + 1 >= n || a.at(i) + b.at(j) <= a.at(i) + a.at(i + 1))

            && (j + 1 >= n || a.at(i) + b.at(j) <= b.at(j) + b.at(j + 1))) {

            b.at(k) = a.at(i) + b.at(j);

            ans += b.at(k);

            i++;

            j++;

            continue;

        }

        if (j + 1 < n && b.at(j + 1) < LLONG\_MAX && (i >= n || b.at(j) + b.at(j + 1) <= b.at(j) + a.at(i))

            && (i + 1 >= n || b.at(j) + b.at(j + 1) <= a.at(i) + a.at(i + 1))) {

            b.at(k) = b.at(j) + b.at(j + 1);

            ans += b.at(k);

            j += 2;

        }

    }

    printf("%lld", ans);

    return 0;

}

B

#ifdef \_MSC\_VER

#define \_CRT\_SECURE\_NO\_WARNINGS

#endif

#include <iostream>

#include <string>

#include <vector>

#include <algorithm>

using namespace std;

int main() {

    freopen("bwt.in", "r", stdin);

    freopen("bwt.out", "w", stdout);

    string s;

    vector<string> a;

    cin >> s;

    for (int i = 0; i < s.length(); i++) {

        a.push\_back(s);

        rotate(a.at(i).begin(), a.at(i).begin() + i, a.at(i).end());

    }

    sort(a.begin(), a.end());

    for (int i = 0; i < s.length(); i++) {

        printf("%c", a.at(i)[s.length() - 1]);

    }

    return 0;

}

C

#ifdef \_MSC\_VER

#define \_CRT\_SECURE\_NO\_WARNINGS

#endif

#include <iostream>

#include <string>

#include <vector>

#include <map>

#include <algorithm>

using namespace std;

int main() {

    freopen("mtf.in", "r", stdin);

    freopen("mtf.out", "w", stdout);

    string s;

    cin >> s;

    vector<int> ans;

    map<char, int> key;

    int min\_key = 0;

    int difference = 1;

    for (char i = 'a'; i <= 'z'; i++) {

        key.emplace(i, i - 'a');

    }

    for (char c : s) {

        ans.push\_back(1 + key.at(c));

        for (char i = 'a'; i <= 'z'; i++) {

            if (key.at(i) < key.at(c)) {

                key.at(i)++;

            }

        }

        key.at(c) = 0;

    }

    for (int i : ans) {

        printf("%i ", i);

    }

    return 0;

}

D

#ifdef \_MSC\_VER

#define \_CRT\_SECURE\_NO\_WARNINGS

#endif

#include <iostream>

#include <cstring>

#include <string>

#include <vector>

#include <map>

#include <algorithm>

using namespace std;

int main() {

    freopen("lzw.in", "r", stdin);

    freopen("lzw.out", "w", stdout);

    string s;

    vector <int> t;

    map<string, int> key;

    for (char i = 'a'; i <= 'z'; i++) {

        key.emplace(string (1, i), i - 'a');

    }

    cin >> s;

    int i = 0, j, max = 25;

    while (i < s.size()) {

        j = 1;

        while (i + j <= s.size() && key.find(s.substr(i, j)) != key.end()) {

            j++;

        }

        t.push\_back(key.at(s.substr(i, j - 1)));

        if (i + j - 1 < s.size()) {

            key.emplace(s.substr(i, j), ++max);

            i += j - 1;

        }

        else {

            break;

        }

    }

    for (int i : t) {

        printf("%i ", i);

    }

    return 0;

}