**KMP**

void kmp() {

cin >> (A + 1);

cin >> (B + 1);

n = strlen(A + 1);

m = strlen(B + 1);

for (int i = 2, j = 0; i <= m; i++) {

while (j && B[i] != B[j + 1]) j = ne[j];

if (B[i] == B[j + 1]) {

j++;

}

ne[i] = j;

}

for (int i = 1, j = 0; i <= n; i++) {

while (j && A[i] != B[j + 1]) j = ne[j];

if (A[i] == B[j + 1]) {

j++;

}

if (j == m) {

j = ne[j];

}

}

}

**EXKMP**

void exkmp() {

cin >> (s1 + 1) >> (s2 + 1);

n = strlen(s1 + 1);

m = strlen(s2 + 1);

for (int i = 1; i <= n; i++) {

s[i] = s1[i];

}

s[n + 1] = '#';

for (int i = 1; i <= m; i++) {

s[n + i + 1] = s2[i];

}

z[1] = 0;

int L = 1, R = 0;

for (int i = 2; i <= n + m + 1; i++) {

if (i > R) {

z[i] = 0;

} else {

int k = i - L + 1;

z[i] = min(z[k], R - i + 1);

}

while (z[i] + i <= n + m + 1 && s[z[i] + 1] == s[z[i] + i]) {

++z[i];

}

if (i + z[i] - 1 > R) {

R = i + z[i] - 1;

L = i;

}

}

}

**MANACHER**

void manacher() {

n = strlen(a + 1);

m = 0;

b[++m] = '#';

for (int i = 1; i <= n; i++) {

b[++m] = a[i];

b[++m] = '#';

}

int M = 0, R = 0;

for (int i = 1; i <= m; i++) {

if (i > R) {

p[i] = 1;

} else {

p[i] = min(p[2 \* M - i], R - i + 1);

}

while (i - p[i] >= 1 && i + p[i] <= m && b[i - p[i]] == b[i + p[i]]) {

++p[i];

}

if (i + p[i] - 1 > R) {

R = i + p[i] - 1;

M = i;

}

}

int ans = 0;

for (int i = 1; i <= m; i++) {

ans = max(ans, p[i]);

}

cout << ans - 1 << endl;

}

AC自动机

int tr[N][26], idx，fail[N]，q[N], cnt[N];

void insert() {

int p = 0;

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

int t = str[i] - 'a';

if (tr[p][t] == 0) tr[p][t] = ++idx;

p = tr[p][t];

}

cnt[p]++;

}

void build() {

int hh = 0, tt = -1;

for (int i = 0; i < 26; i++) {

if (tr[0][i] != 0) {

q[++tt] = tr[0][i];

}

}

while (hh <= tt) {

int t = q[hh++];

for (int k = 0; k < 26; k++) {

if (tr[t][k] == 0) {

tr[t][k] = tr[fail[t]][k];

} else {

fail[tr[t][k]] = tr[fail[t]][k];

q[++tt] = tr[t][k];

}

}

}

}

int main() {

cin >> n;

for (int i = 0; i < n; i++) {cin >> str; insert();}

build(); cin >> str;

int res = 0;

for (int i = 0, j = 0; str[i]; i++) {

int t = str[i] - 'a';

j = tr[j][t];

int p = j;

while (p && cnt[p] != -1) {

res += cnt[p];

cnt[p] = -1;

}

}

cout << res << endl;

}