

Contents

1 String

1.1 SAM

```

class SAM{
private:
    struct node{
        int ch[26];
        int len, pa, t, chd;
        bool is_pre;
        node() {
            memset(ch, 0, sizeof(ch));
            len = pa = t = chd = 0;
            is_pre = 0;
        }
    } arr[MAXN<<1];
    vector<int> reBFS[MAXN];
    int cnt, las;
    void add(int c) {
        int p = las;
        int cur = las = ++cnt;
        arr[cur].len = arr[p].len + 1;
        arr[cur].is_pre = 1;
        while (p && !arr[p].ch[c]) {
            arr[p].ch[c] = cur;
            p = arr[p].pa;
        }
        if (!arr[p].ch[c]) {
            arr[cur].pa = 0;
            arr[0].chd++;
            arr[p].ch[c] = cur;
        } else {
            int q = arr[p].ch[c];
            if (arr[q].len == arr[p].len + 1) {
                arr[cur].pa = q;
                arr[q].chd++;
            } else {
                int nq = ++cnt;
                arr[nq] = arr[q];
                arr[nq].is_pre = 0;
                arr[nq].len = arr[p].len + 1;
                arr[q].pa = arr[cur].pa = nq;
                arr[nq].chd = 2;
                for (; arr[p].ch[c] == q; p = arr[p].pa)
                    arr[p].ch[c] = nq;
            }
        }
    }
public:
    void init(string s) {
        for (int i = 0; i <= cnt; i++)
            arr[i] = node();
        cnt = las = 0;
        arr[0].t = 1;
        for (int i = 0; i < s.size(); i++)
            add(s[i] - 'a');
        queue<int> que;
        for (int i = 1; i <= cnt; i++)
            if (!arr[i].chd) que.push(i);
        while (que.size()) {
            int now = que.front();
            que.pop();
            if (arr[now].is_pre) arr[now].t++;
            arr[arr[now].pa].t += arr[now].t;
            arr[arr[now].pa].chd--;
            if (arr[now].pa && !arr[arr[now].pa].chd)
                que.push(arr[now].pa);
        }
    }
    int solve(string &p) {
        int now = 0;
        for (int i = 0; i < p.size(); i++) {
            if (arr[now].ch[p[i] - 'a'])
                now = arr[now].ch[p[i] - 'a'];
            else return 0;
        }
        return arr[now].t;
    }
};

```