

时间复杂度优化相关题目练习

人员

温郝冬、罗启宸、董浩桢、陶汇笙、郭栩睿、马瑞昕、崔宸赫、洪晨栋、洪晨棋 到课

作业检查

温郝冬、罗启宸、郭栩睿、陶汇笙、崔宸赫 全部完成 A、B 题

张昱霖、邹忆航、周苇杰 做了，有个别题目没过

作业

<https://www.luogu.com.cn/contest/192684>

A、B、C、D 4道题

课堂表现

陶汇笙 同学课上做题比较好，提出表扬！！

课堂内容

T465969 change

```
// 暴力做法
#include <bits/stdc++.h>

using namespace std;

int main()
{
    int n;
    string s;
    cin >> n >> s;

    int Q; cin >> Q;
    while (Q -- ) {
        int op, x;
        char c;
        cin >> op >> x >> c;
        if (op == 1) {
            s[x-1] = c;
        }
        else if (op == 2) {
            for (int i = 0; i < n; i++) {
                if (s[i]>='A' && s[i]<='Z') {
                    s[i] = s[i] - 'A' + 'a';
                }
            }
        }
    }
}
```

```

    }
}
else {
    for (int i = 0; i < n; i++) {
        if (s[i] >= 'a' && s[i] <= 'z') {
            s[i] = s[i] - 'a' + 'A';
        }
    }
}
}

for (int i = 0; i < n; i++) {
    cout << s[i];
}
return 0;
}

```

1. 输入 -> 用结构体数组存下来
2. 从后往前找最后一个 2/3, 用 pos 存下来
3. 从前往后做
 - > 如果是 1 操作, 就做
 - > 否则, 看是否是第 pos 个。是的话做, 不是的话不做

```

// 满分做法
#include <bits/stdc++.h>

using namespace std;

struct node {
    int op;
    int x;
    char c;
};

const int maxn = 500000 + 5;

node a[maxn];

int main()
{
    int n;
    string s;
    cin >> n >> s;

    int Q; cin >> Q;
    for (int j = 1; j <= Q; j++) {
        cin >> a[j].op >> a[j].x >> a[j].c;
    }
}

```

```
int pos = -1;
for (int j = Q; j >= 1; j--) {
    if (a[j].op==2 || a[j].op==3) {
        pos = j;
        break;
    }
}

for (int j = 1; j <= Q; j++) {
    int op = a[j].op, x = a[j].x;
    char c = a[j].c;

    if (op == 1) {
        s[x-1] = c;
    }
    else {
        if (j == pos) {
            if (op == 2) {
                for (int i = 0; i < n; i++) {
                    if (s[i]>='A' && s[i]<='Z') {
                        s[i] = s[i]-'A'+'a';
                    }
                }
            }
            else {
                for (int i = 0; i < n; i++) {
                    if (s[i]>='a' && s[i]<='z') {
                        s[i] = s[i]-'a'+'A';
                    }
                }
            }
        }
    }
}

for (int i = 0; i < n; i++) {
    cout << s[i];
}
return 0;
}
```

P5015 [NOIP2018 普及组] 标题统计

```
#include <bits/stdc++.h>

using namespace std;

bool a[300];

int main()
{
```

```
string s;
getline(cin, s);
int n = s.size(), cnt = 0;
for (int i = 0; i < n; ++i) {
    if (s[i] != ' ') ++cnt;
}

cout << cnt << endl;
return 0;
}
```

P1554 梦中的统计

```
#include <bits/stdc++.h>

using namespace std;

int a[15];

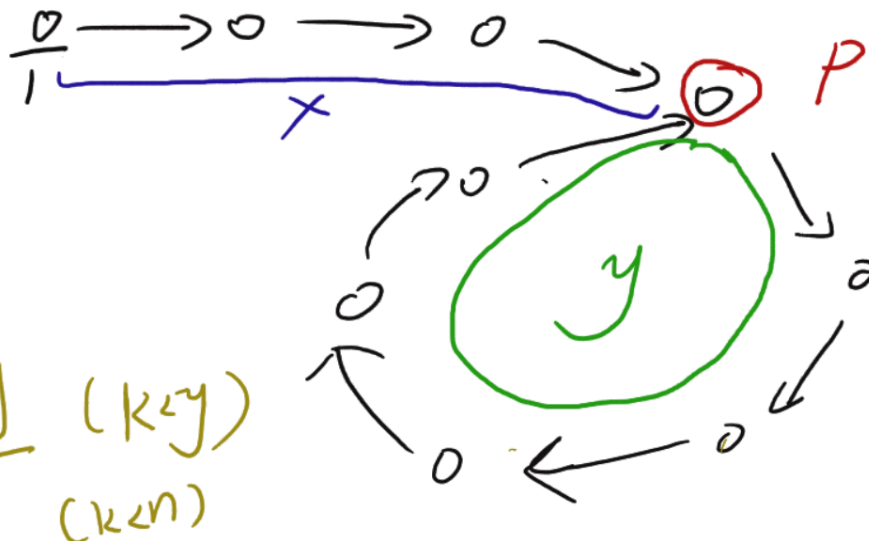
void solve(int x) {
    while (x) {
        int t = x%10;
        ++a[t];
        x /= 10;
    }
}

int main()
{
    int m, n; cin >> m >> n;
    for (int i = m; i <= n; ++i) solve(i);

    for (int i = 0; i <= 9; ++i) cout << a[i] << " ";
    cout << endl;
    return 0;
}
```

T471233 transmit

K



做:

1. $k = x$

2. $k \% y = y$ ($k < y$)
 ($k < n$)

#include <bits/stdc++.h>

using namespace std;

const int maxn = 200000 + 5;

int a[maxn];

int tmm[maxn]; // tm[i]: 到城镇 i 的时间

int main()

{

int n; long long k; cin >> n >> k;

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

cin >> a[i];

}

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

tmm[i] = -1;

}

// 当 tmm[i] == -1 时, 说明没到过 城镇 i

// 当 tmm[i] != -1 时, 说明在 tmm[i] 这个时间到过 城镇 i

int pos = 1;

tmm[1] = 0;

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

pos = a[pos];

// i 时刻, 到了 pos 这个城镇

if (tmm[pos] == -1) {

tmm[pos] = i;

} else {

// 之前在 tmm[pos] 这个时间到过, 现在在 i 时间到过

int p = pos;

```
        int x = tmm[pos];
        int y = i - tmm[pos];

        k -= x;
        k %= y; // 最后要走 k 步
        while (k -- ) {
            p = a[p];
        }
        cout << p << endl;
        return 0;
    }
}
cout << pos << endl;
return 0;
}
```