

百度松果菁英班 · 码蹄集 · OJ周赛

2024年第9周

#模拟

#约数遍历 遍历数 n 的所有约数只需考察 $i, \frac{n}{i}$ 从 $i = 1$ 到 \sqrt{n} (即 $i \leq \frac{n}{i}$) , 复杂度 $O(\sqrt{n})$

2024年第10周

#计算几何

#平面最近点对

分治法: <https://oi-wiki.org/geometry/nearest-points/>

非分治: <https://oi-wiki.org/geometry/nearest-points/>

#染色问题

横竖染色, 分治, `ans=min(heng+sum(shu1),shu0)`

#斐波拉契数列

$$\gcd(\text{fib}(x), \text{fib}(y)) = \text{fib}(\gcd(x, y))$$

2024年第11周

#二分答案

问题不方便正向求解, 则宁愿多乘 $O(\log n)$ 的复杂度进行二分法验证可能的答案, 逐渐逼近 `false` 区间与 `true` 区间的交界处。

#前缀和 #位运算

考察每一位的情况, 利用满足题目要求的条件限制进行分析。**正难则反**, 也可以考察不满足题意的一些情况。由于各位对答案的作用有区别, 我们从影响最大的开始, 同时不断施加对后续影响较小的的限制。

注意: `1<<60` 这样的代码是错误的, 溢出了, 应该写成 `1ll<<60`。

2024年第12周

#ST表

基于倍增思想。建立 $O(n \log n)$, 查询 $O(1)$, 建立后不可修改。

```

const int L = 22;
int Log[N], ma[N][L], mi[N][L];
inline void init() {
    Log[1] = 0;
    Log[2] = 1;
    for (int i = 3; i < N; ++i)
        Log[i] = Log[i >> 1] + 1;

    for (int i = 1; i ≤ n; ++i) {
        ma[i][0] = a[i];
        mi[i][0] = b[i];
    }
    for (int j = 1; j ≤ L; ++j)
        for (int i = 1; i + (1 << j) - 1 ≤ n; ++i) {
            ma[i][j] = max(ma[i][j - 1], ma[i + (1 << j - 1)][j - 1]);
            mi[i][j] = min(mi[i][j - 1], mi[i + (1 << j - 1)][j - 1]);
        }
}

inline int qry_max(int l, int r) {
    int q = Log[r - l + 1];
    return max(ma[l][q], ma[r - (1 << q) + 1][q]);
}

inline int qry_min(int l, int r) {
    int q = Log[r - l + 1];
    return min(mi[l][q], mi[r - (1 << q) + 1][q]);
}

```

#田忌赛马

尽力得分：强vs强，弱vs弱，比不过就用弱送人头换对方最强，然后又强vs强，弱vs弱……循环往复。

#全错位排列数

递推关系： $D_n = (n - 1)(D_{n-1} + D_{n-2})$, $D_1 = 0$, $D_2 = 1$

2024年第13周

#贪心

带反悔的贪心，用 `priority_queue` 维护当前选择，后面不断判断是否需要替换堆顶元素。

[MT3027 red and blue](#)

[MT3028 正反卡牌](#)