

# 综合练习

---

## 人员

赵熙羽、杨瑾硕、谢亚锴、陈洛冉、于子珈、隋天乙、司云心、刘闯速、牛同泽、程梓豪、秦显森、孙靖轲 到课

## 作业

<https://cppoj.kids123code.com/contest/754> (课上讲了 A ~ D 题, 课后作业是 E 题)

## 课堂表现

今天的前两道题, 分别是去年 CSP-S 的第一题 和 CSP-X 的第四题, 这两道题对现在的同学们来说都不应该是难题才对

但是同学们这两道题课上做的都不太好, 在老师讲解之前都不太会做, 课下要好好复习一下这几道题目

## 课堂内容

### [CSP-S 2024] 决斗

先排序, 然后双指针扫一遍

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

using namespace std;

const int maxn = 1e5 + 5;
int w[maxn];

int main()
{
    int n; cin >> n;
    for (int i = 1; i <= n; ++i) cin >> w[i];
    sort(w+1, w+n+1);

    int res = n;
    for (int i = 2, j = 1; i <= n; ++i) {
        if (w[i] > w[j]) --res, ++j;
    }
    cout << res << endl;
    return 0;
}
```

### [CSP-X2024 山东] 刷题

二分一个 mid, 看一天最大时间是 mid 时, 能否让总天数  $\leq m$

```

#include <bits/stdc++.h>

using namespace std;

const int maxn = 1e5 + 5;
int w[maxn];
int n, m;

bool check(int mid) {
    int cnt = 1, maxx = w[1], sum = w[1];
    for (int i = 2; i <= n; ++i) {
        int new_maxx = max(maxx, w[i]), new_sum = sum + w[i];
        if (new_sum - new_maxx > mid) {
            cnt++, maxx = w[i], sum = w[i];
        } else {
            maxx = new_maxx, sum = new_sum;
        }
    }
    return cnt <= m;
}

int main()
{
    cin >> n >> m;
    for (int i = 1; i <= n; ++i) cin >> w[i];

    int l = 0, r = 1e9;
    while (l <= r) {
        int mid = (l + r) / 2;
        if (check(mid)) r = mid - 1;
        else l = mid + 1;
    }
    cout << l << endl;
    return 0;
}

```

## Count Bracket Sequences

dp, 设  $f[i][j]$  代表以第  $i$  位结尾,  $1 \sim i$  中  $(-) = j$  时的方案数

答案是  $f[n][0]$ , 初值是  $f[0][0] = 1$

转移分为 左括号转移 和 右括号转移 两种情况

```

#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int maxn = 3000 + 5;

```

```

const int mod = 998244353;
char s[maxn];
int f[maxn][maxn]; // f[i][j]: 以第 i 位结尾, 1~i 中 ( - ) ==j 时的方案数

int main()
{
    cin >> (s+1);
    int n = strlen(s+1);

    f[0][0] = 1;
    for (int i = 1; i <= n; ++i) {
        for (int j = 0; j <= i; ++j) {
            if (s[i]=='(' || s[i]=='?') {
                if (j > 0) f[i][j] = f[i-1][j-1];
            }
            if (s[i]==')' || s[i]=='?') {
                f[i][j] = (f[i][j] + f[i-1][j+1]) % mod;
            }
        }
    }

    if (s[n] == '(') cout << 0 << endl;
    else cout << f[n][0] << endl;
    return 0;
}

```

## [CSP-J 2024] 小木棍

打表找规律

求一个  $f[i]$  数组, 其中  $f[i]$  代表如果有  $i$  个小木棒时, 能表示的最小的数是多少

如何求  $f[i]$  数组, 可以先求一个  $a$  数组, 其中  $a[i]$  代表  $i$  这个数需要用多少根小木棒, 然后利用  $a$  数组就可以求  $f$  数组了

$f$  数组求完之后, 就可以打表找规律了

```

#include <bits/stdc++.h>

using namespace std;

void print(int x) {
    while (x -- ) cout << 8;
    cout << endl;
}

void solve() {
    int n; cin >> n;
    int k = n/7;

    if (n%7 == 1) {

```

```
        if (n == 1) cout << -1 << endl;
        else if (n == 8) cout << 10 << endl;
        else {
            cout << 10;
            print(k-1);
        }
    } else if (n%7 == 2) {
        if (n == 2) cout << 1 << endl;
        else {
            cout << 1;
            print(k);
        }
    } else if (n%7 == 3) {
        if (n == 3) cout << 7 << endl;
        else if (n == 10) cout << 22 << endl;
        else {
            cout << 200;
            print(k-2);
        }
    } else if (n%7 == 4) {
        if (n == 4) cout << 4 << endl;
        else {
            cout << 20;
            print(k-1);
        }
    } else if (n%7 == 5) {
        cout << 2;
        print(k);
    } else if (n%7 == 6) {
        cout << 6;
        print(k);
    } else {
        print(k);
    }
}

int main()
{
    int T; cin >> T;
    while (T -- ) solve();
    return 0;
}
```