杂题混练

人员

于珈浩、赵清航、左子毅、刘佳赫、孙乐涵、刘子淇、周子航、杨洋 到课

作业检查

上周作业链接：https://vjudge.net/contest/652462

密码：code@123

孙乐涵 完成 2 道题

杨洋、刘子淇 完成 1 道题

其他同学未做

作业

https://vjudge.net/contest/652462, 上周 5 道作业题要求大家补完

https://www.luogu.com.cn/contest/197227, 课上 1 道题要求大家补完

https://vjudge.net/contest/653870, 课后作业 5 道题要求大家课后进行思考尝试

课堂表现

课上整体纪律比较好，同学们课下要积极进行补题。

课堂内容

CF1119D Frets On Fire

二分查找

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

using namespace std;

typedef long long LL;
const int maxn = 1e5 + 5;
LL w[maxn], a[maxn], pSum[maxn];

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

```
  --n;
  for (int i = 1; i <= n; ++i) a[i] = w[i+1] - w[i];
  sort(a+1, a+n+1);
  for (int i = 1; i <= n; ++i) pSum[i] = pSum[i-1] + a[i];

  int m; cin >> m;
  while (m -- ) {
    LL l, r; cin >> l >> r;
    LL d = r - l + 1;
    int pos = lower_bound(a+1, a+n+1, d) - a;
    cout << pSum[pos-1] + ((LL)n-pos+2) * d << " ";
  }
  return 0;
}
```

CF1551E Fixed Points

dp

f[i][j]: 处理到第 i 个数时, 保留 j 个数, 最多有几个地方满足 bi=i 的性质

f[i][j] = max(f[i-1][j-1]+(w[i]==j) -> 保留第 i 个数 f[i-1][j] -> 删除第 i 个数)

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

using namespace std;

const int maxn = 2000 + 5;
int w[maxn], f[maxn][maxn];

void solve() {
  int n, k; cin >> n >> k;
  for (int i = 0; i <= n+2; ++i) {
    for (int j = 0; j <= n+2; ++j) f[i][j] = 0;
  }

  for (int i = 1; i <= n; ++i) cin >> w[i];
  for (int i = 1; i <= n; ++i) {
    for (int j = 1; j <= i; ++j) f[i][j] = max(f[i-1][j-1]+(w[i]==j), f[i-1][j]);
  }

  for (int i = n; i >= 1; --i) {
    if (f[n][i] >= k) {
      cout << n-i << endl;
      return;
    }
  }
  cout << -1 << endl;
}

int main()
```

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

CF1901D Yet Another Monster Fight

维护前后缀最大值

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

using namespace std;

const int maxn = 3e5 + 5;
int w[maxn], a[maxn], b[maxn];
int pMax[maxn], sMax[maxn];

int main()
{
  int n; cin >> n;
  for (int i = 1; i <= n; ++i) cin >> w[i];
  for (int i = 1; i <= n; ++i) {
    a[i] = w[i] + n-i, b[i] = w[i] + i-1;
    pMax[i] = max(pMax[i-1], a[i]);
  }
  for (int i = n; i >= 1; --i) sMax[i] = max(sMax[i+1], b[i]);

  int res = 2e9;
  for (int i = 1; i <= n; ++i) {
    res = min(res, max({pMax[i-1], w[i], sMax[i+1]}));
  }
  cout << res << endl;
  return 0;
}
```

AT_arc100_b Equal Cut

双指针 或 二分

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

using namespace std;

typedef long long LL;
const int maxn = 2e5 + 5;
int w[maxn];
LL pSum[maxn];
```

```
LL get_sum(int l, int r) { return pSum[r] - pSum[l-1]; }

LL diff_value(int l, int mid, int r) {
  return abs(get_sum(l, mid) - get_sum(mid+1, r));
}

bool is_better(int l, int mid, int r) {
  return diff_value(l, mid+1, r) <= diff_value(l, mid, r);
}

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

  LL res = 1e18;
  for (int i=1,j=2,k=3; j<=n-2; ++j) {
    while (i+1<j && is_better(1,i,j)) ++i;
    while (k+1<n && is_better(j+1,k,n)) ++k;
    LL a = get_sum(1, i), b = get_sum(i+1, j), c = get_sum(j+1, k), d =
get_sum(k+1, n);
    res = min(res, max({a,b,c,d}) - min({a,b,c,d}));
  }
  cout << res << endl;
  return 0;
}
```

CF30C Shooting Gallery

根据时间排序，抽象成一张图，递推

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

using namespace std;

typedef long long LL;
const int maxn = 1000 + 5;
struct node {
  int x, y, t; double p;
} w[maxn];
vector<int> vec[maxn];
bool cmp(node a, node b) { return a.t < b.t; }

LL dis_2(int x1, int y1, int x2, int y2) {
  int dx = x1-x2, dy = y1-y2;
  return (LL)dx*dx + (LL)dy*dy;
}
double f[maxn];
```

```
int main()
{
  int n; cin >> n;
  for (int i = 1; i <= n; ++i) cin >> w[i].x >> w[i].y >> w[i].t >> w[i].p;
  sort(w+1, w+n+1, cmp);

  for (int i = 1; i <= n; ++i) {
    for (int j = i+1; j <= n; ++j) {
      int d2 = dis_2(w[i].x, w[i].y, w[j].x, w[j].y);
      int t = w[j].t-w[i].t;
      if (d2 <= (LL)t*t) vec[j].push_back(i);
    }
  }

  double res = 0;
  for (int i = 1; i <= n; ++i) {
    for (int j : vec[i]) f[i] = max(f[i], f[j]);
    f[i] += w[i].p;
    res = max(res, f[i]);
  }
  printf("%.8f\n", res);
  return 0;
}
```

T494829 box

f[i][j]: 至少有 i 个章鱼小丸子、j 个大饼时, 最少需要多少饭盒

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

using namespace std;

const int maxn = 300 + 5;
const int inf = 0x3f3f3f3f;
int f[maxn][maxn];

int main()
{
  int n; cin >> n;
  int x, y; cin >> x >> y;

  memset(f, 0x3f, sizeof(f));
  f[0][0] = 0;

  while (n -- ) {
    int a, b; cin >> a >> b;
    for (int i = x; i >= 0; --i) {
      for (int j = y; j >= 0; --j) f[i][j] = min(f[i][j], f[max(0,i-a)][max(0,j-
b)]+1);
    }
  }
```

```
    cout << (f[x][y]==inf?-1:f[x][y]) << endl;
    return 0;
}
```