# 杂题练习

## 人员

左子毅、朱奕鸣、杨洋、赵清航、于珈浩 到课

## 作业检查

左子毅 已完成

朱奕鸣 已完成

杨洋 已完成

赵清航 上周请假

王崇宇 已完成

于珈浩 已完成

刘佳赫 已完成

## 作业

https://www.luogu.com.cn/contest/177241, 课上题目要求补完

https://www.luogu.com.cn/contest/176989，这些是课下作业题，要求同学们课下思考实现，下节课讲

## 课堂表现

同学们在做题时还是会有各种各样的问题，比如：不开 long long，读题不清晰，做题策略问题 等等

以后同学们一定要尽可能细心

## 课堂内容

### CF1828A Divisible Array

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

using namespace std;

void solve() {
  int n; cin >> n;
  for (int i = 1; i <= n; ++i) cout << i*2 << " ";
  cout << endl;
}

int main()
{
```

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

## CF1860B Fancy Coins

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

using namespace std;

typedef long long LL;

void solve() {
  int m, k, a1, ak; cin >> m >> k >> a1 >> ak;
  int res = 0;
  int x = m/k, y = m%k;

  if (a1 >= y) a1 -= y;
  else res += y-a1, a1 = 0;

  ak += a1/k;
  res += max(0, x-ak);
  cout << res << endl;
}

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

## CF1272D Remove One Element

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

using namespace std;

typedef long long LL;
const int maxn = 2e5 + 5;
int w[maxn], f[maxn], s[maxn];

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

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

```
      if (w[i] > w[i-1]) f[i] = f[i-1] + 1;
      else f[i] = 1;
    }
    for (int i = n; i >= 1; --i) {
      if (w[i] < w[i+1]) s[i] = s[i+1] + 1;
      else s[i] = 1;
    }

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

int main()
{
    int T = 1;
    while (T -- ) solve();
    return 0;
}
```

## CF1198A MP3

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

using namespace std;

typedef long long LL;
const int maxn = 4e5 + 5;
int w[maxn];

void solve() {
    int n, I; cin >> n >> I;
    for (int i = 1; i <= n; ++i) cin >> w[i];
    I = (I*8) / n;
    if (I >= 20) { cout << 0 << endl; return; }

    sort(w+1, w+n+1);

    int limit = (1<<I), res = 0;
    map<int, int> mp;
    for (int l=1, r=0; l <= n; ++l) {
      r = max(r, l-1);
      while (r+1<=n && ((int)mp.size()<limit || mp.count(w[r+1]))) {
        mp[w[r+1]]++;
        ++r;
      }
```

```
      res = max(res, r-l+1);
      mp[w[l]]--;
      if (mp[w[l]] == 0) mp.erase(w[l]);
    }

    cout << n-res << endl;
}

int main()
{
    int T = 1;
    while (T -- ) solve();
    return 0;
}
```

### AT_abc290_e [ABC290E] Make it Palindrome

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

using namespace std;

typedef long long LL;
const int maxn = 2e5 + 5;
int w[maxn];
vector<int> vec[maxn];

LL calc(int n, vector<int> vv) {
    LL res = 0;
    int len = vv.size();
    for (int i = 0, j = len-1; i < j; ) {
        int x = vv[i], y = vv[j];
        int lDis = x, rDis = n-y+1;
        if (lDis <= rDis) res += (LL)lDis * (j-i), ++i;
        else res += (LL)rDis * (j-i), --j;
    }
    return res;
}

int main()
{
    int n; cin >> n;
    for (int i = 1; i <= n; ++i) {
        cin >> w[i];
        vec[w[i]].push_back(i);
    }

    LL res = 0;
    for (int i = 2; i <= n; ++i) res += ((LL)n-i+1) * (i/2);

    for (int i = 1; i <= n; ++i) res -= calc(n, vec[i]);
```

```cpp
    cout << res << endl;
    return 0;
}
```