杂题混练 + 质数筛

人员

蔡云翔、石宇嫿、李佳声、窦浩轩、胡赫轩、崔嘉睿、穆鹏宇、程晟泰、梁钰涵、王梓同、刘浩、纪昀琨、张彧韶、王景睿、朱奕鸣、孙乐涵、于潇涵、王雨辰 到课, 刘浩 看录屏

作业检查

上周作业链接：https://vjudge.net/contest/653870

密码：code@123

张彧韶、穆鹏宇、胡赫轩 做了 4 道题

梁钰涵、纪昀琨、程晟泰 做了 2 道题

孙乐涵、崔嘉睿 做了 1 道题

其他同学 未做

作业

https://vjudge.net/contest/653870, 上周 5 道作业题要求大家补完

https://www.luogu.com.cn/contest/198316, 课上 K 题要求大家补完

https://vjudge.net/contest/655631, 课后作业 5 道题要求大家课后进行思考尝试

课堂表现

有些同学课上听讲没问题，课后补题有点跟不上了，这样只会越拖越多，一定要挤时间补题。

课堂内容

CF1896D Ones and Twos

```cpp
// 二分 + 树状数组写法
#include <bits/stdc++.h>

using namespace std;

void print(bool flag) { cout << (flag?"YES":"NO") << endl; }

const int maxn = 1e5 + 5;
int n, w[maxn], tr[maxn];
int lowbit(int x) { return x & (-x); }
void update(int x, int k) {
  while (x <= n) { tr[x] += k, x += lowbit(x); }
}
```

```
int query(int x) {
  int res = 0;
  while (x) { res += tr[x]; x -= lowbit(x); }
  return res;
}

int lquery() {
  int l = 0, r = n;
  while (l <= r) {
    int mid = (l + r) / 2;
    if (query(mid) == 2*mid) l = mid+1;
    else r = mid-1;
  }
  return r;
}
int rquery(int sum) {
  int l = 0, r = n;
  while (l <= r) {
    int mid = (l + r) / 2;
    if (sum - query(mid) == 2 * (n-mid)) r = mid-1;
    else l = mid+1;
  }
  return n-l;
}

void solve() {
  int m; cin >> n >> m;
  for (int i = 0; i <= n+2; ++i) tr[i] = 0;

  int sum = 0;
  for (int i = 1; i <= n; ++i) {
    cin >> w[i]; update(i, w[i]); sum += w[i];
  }

  while (m -- ) {
    int op; cin >> op;
    if (op == 1) {
      int s; cin >> s;
      if (s > sum) { print(false); continue; }
      if ((sum-s) % 2 == 0) { print(true); continue; }
      int k = (sum-s+1) / 2;
      if (min(lquery(), rquery(sum)) >= k) print(false);
      else print(true);
    } else {
      int i, v; cin >> i >> v;
      update(i, -w[i]); sum -= w[i];
      w[i] = v;
      update(i, w[i]); sum += w[i];
    }
  }
}

int main()
{
```

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

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

using namespace std;

void print(bool flag) { cout << (flag?"YES":"NO") << endl; }

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

int lquery(set<int>& s) {
  if (s.empty()) return n;
  int p = *s.begin();
  return p-1;
}

int rquery(set<int>& s) {
  if (s.empty()) return n;
  int p = *s.rbegin();
  return n - p;
}

void solve() {
  int m; cin >> n >> m;

  set<int> st;
  int sum = 0;
  for (int i = 1; i <= n; ++i) {
    cin >> w[i]; sum += w[i];
    if (w[i] == 1) st.insert(i);
  }

  while (m -- ) {
    int op; cin >> op;
    if (op == 1) {
      int s; cin >> s;
      if (s > sum) { print(false); continue; }
      if ((sum-s) % 2 == 0) { print(true); continue; }
      int k = (sum-s+1) / 2;
      if (min(lquery(st), rquery(st)) >= k) print(false);
      else print(true);
    } else {
      int i, v; cin >> i >> v;
      sum -= w[i]; if (w[i] == 1) st.erase(i);
      w[i] = v;
      sum += w[i]; if (w[i] == 1) st.insert(i);
    }
```

```
    }
}

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

CF1280A Cut and Paste

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

using namespace std;

typedef long long LL;
const int mod = 1e9+7;

void solve() {
    int n; string s; cin >> n >> s;
    bool flag = false; int tot = n;
    for (int i = 0; i < n; ++i) {
        int k = s[i]-'0';
        if (!flag) {
            string r = s.substr(i+1);
            --k;
            while (k -- ) s += r;
            tot = s.size();
            if (tot > n) flag = true;
        } else {
            int rlen = (tot - i - 1 + mod) % mod;
            tot = (tot + ((LL)k-1)*rlen) % mod;
        }
    }

    cout << tot << endl;
}

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

CF14E Camels

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

using namespace std;

int f[25][5][5][15][15];
// f[i][j][k][a][b]: 第 i 个数为 j，第 i-1 个数为 k 时，前面有 a 个升峰，有 b 个降峰时的方案数

int main()
{
  int n, t; cin >> n >> t;
  for (int j = 1; j <= 4; ++j) {
    for (int k = 1; k <= 4; ++k) {
      if (j==k) continue;
      f[2][j][k][0][0]++;
    }
  }

  for (int i = 3; i <= n; ++i) {
    for (int j = 1; j <= 4; ++j) {  // i
      for (int k = 1; k <= 4; ++k) {  // i-1
        for (int l = 1; l <= 4; ++l) {  // i-2
          for (int a = 0; a <= t; ++a) {
            for (int b = 0; b <= t-1; ++b) {
              if (j==k || k==l) continue;
              if (j>k && l>k) {
                if (b) f[i][j][k][a][b] += f[i-1][k][l][a][b-1];
              }
              else if (j<k && l<k) {
                if (a) f[i][j][k][a][b] += f[i-1][k][l][a-1][b];
              }
              else {
                f[i][j][k][a][b] += f[i-1][k][l][a][b];
              }
            }
          }
        }
      }
    }
  }

  int res = 0;
  for (int j = 1; j <= 4; ++j) {
    for (int k = 1; k <= 4; ++k) {
      res += f[n][j][k][t][t-1];
    }
  }
  cout << res << endl;
  return 0;
}
```

CF3B Lorry

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

using namespace std;

const int maxn = 1e5 + 5;
struct node {
  int t, p;
  int id;
} w[maxn];
bool cmp(node a, node b) {
  if (a.t == 1) a.p *= 2;
  if (b.t == 1) b.p *= 2;
  return a.p > b.p;
}
bool st[maxn];

int main()
{
  int n, v; cin >> n >> v;
  for (int i = 1; i <= n; ++i) {
    int t, p; cin >> t >> p; w[i] = {t, p, i};
  }

  sort(w+1, w+n+1, cmp);
  vector<node> vec1;
  for (int i = 1; i <= n; ++i) {
    if (w[i].t == 1) vec1.push_back(w[i]);
  }

  int res = 0;
  for (int i = 1, j = 0, k = 0; i <= n; ++i) {
    if (v >= w[i].t) {
      v -= w[i].t; res += w[i].p; st[w[i].id] = true;
      if (w[i].t == 1) ++j; else ++k;
    } else if (v == 1) {
      if (j == 0) continue;
      node x = vec1[j - 1];
      node y = {0, 0, 0};
      if (j < (int)vec1.size()) y = vec1[j];
      if (w[i].p > x.p+y.p) {
        res -= x.p; st[x.id] = false;
        res += w[i].p; st[w[i].id] = true;
        break;
      } else {
        res += y.p; st[y.id] = true;
        break;
      }
    } else break;
  }
```

```
  cout << res << endl;
  for (int i = 1; i <= n; ++i) {
    if (st[i]) cout << i << " ";
  }
  cout << endl;
  return 0;
}
```

CF12E Start of the season

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

using namespace std;

const int maxn = 1000 + 5;
int w[maxn][maxn];

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

  for (int i = 1; i <= n-1; ++i) {
    w[i][n] = w[i][i], w[i][i] = 0;
    w[n][i] = w[i][n];
  }

  for (int i = 1; i <= n; ++i) {
    for (int j = 1; j <= n; ++j) cout << w[i][j] << " ";
    cout << endl;
  }
  return 0;
}
```

P5736 【深基7.例2】质数筛

埃氏筛 O(n * loglogn)

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

using namespace std;

const int maxn = 1e5 + 5;
bool f[maxn];
```

```
int main()
{
  for (int i = 2; i < maxn; ++i) {
    if (f[i]) continue;
    for (int j = i+i; j < maxn; j += i) f[j] = true;
  }

  int n; cin >> n;
  while (n -- ) {
    int x; cin >> x;
    if (!f[x] && x!=1) cout << x << " ";
  }
  return 0;
}
```

欧拉筛 O(n)

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

using namespace std;

const int maxn = 1e5 + 5;
bool f[maxn];
vector<int> primes;

int main()
{
  for (int i = 2; i < maxn; ++i) {
    if (!f[i]) primes.push_back(i);
    for (int j : primes) {
      if (i * j >= maxn) break;
      f[i * j] = true;
      if (i % j == 0) break;
    }
  }

  int n; cin >> n;
  while (n -- ) {
    int x; cin >> x;
    if (!f[x] && x!=1) cout << x << " ";
  }
  return 0;
}
```