综合混练

人员

牛晓晨、刘新睿、隋天翼、范家郡 到课

上周作业检查

上周作业链接: https://vjudge.net/contest/719073

作业

https://vjudge.net/contest/720521 (课上讲了上周比赛的 A B C D E, 课后作业是本周比赛的 A B C D E 题)

课堂表现

同学们课上整体听讲很认真, 补题情况也整体不错, 对同学们提出表扬！！

课堂内容

CF1512F Education

枚举以每个数作为结尾时需要多少天, 对所有情况求最小值

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

using namespace std;

typedef long long LL;
const int maxn = 2e5 + 5;
int a[maxn], b[maxn];
```

```
int get_up(int a, int b) { return (a<=0 ? 0 : (a+b-1) / b); }

void solve() {
  int n, c; cin >> n >> c;
  for (int i = 1; i <= n; ++i) cin >> a[i];
  for (int i = 1; i <= n-1; ++i) cin >> b[i];

  LL res = 1e18;
  LL day = 0, sum = 0;
  for (int i = 1; i <= n; ++i) {
    res = min(res, day + get_up(c-sum, a[i]));  // 以 i 结尾
    if (i == n) break;

    int k = get_up(b[i]-sum, a[i]);  // k 天才能凑够 b[i]
    day += k+1, sum = sum + k*a[i] - b[i];
  }

//  cout << "----------------------- ";
  cout << res << endl;
}

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

CF1485C Floor and Mod

设 k 为 a/b, 也是 a%b

那么就是 a = k*b + k (k < b)

枚举 k, 对应的 b 的范围是 [k+1, min(y, (x-k)/k)]

k 的枚举范围为 [1, min(y-1, sqrt(x))]

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

using namespace std;

typedef long long LL;

void solve() {
  int x, y; cin >> x >> y;
  LL res = 0;
  for (int k = 1; k <= y-1; ++k) {
    int l = k+1, r = min((x-k)/k, y);
    if (r < l) break;
    res += r-l+1;
  }
```

```
//  cout << "---------------------- ";
  cout << res << endl;
}

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

CF1902E Collapsing Strings

构建一棵 trie 存储所有输入的字符串, 然后遍历每个字符串, 翻转这个字符串后在 trie 树中进行查询即可

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

using namespace std;

int get_int(char x) { return x - 'a'; }

typedef long long LL;
const int N = 1e6 + 5, M = 26;
int tr[N][M], idx = 0;
int f[N];

void tr_insert(string s) {
  int p = 0;
  for (char i : s) {
    int u = get_int(i);
    if (!tr[p][u]) tr[p][u] = ++idx;
    p = tr[p][u];
    ++f[p];
  }
}

LL tr_query(string s) {
  int p = 0; LL res = 0;
  for (char i : s) {
    int u = get_int(i);
    if (!tr[p][u]) break;
    p = tr[p][u];
    res += f[p];
  }
  return res;
}

string str[N];

int main()
```

```
{
  ios::sync_with_stdio(false);
  cin.tie(0);

  int n, len = 0; cin >> n;
  for (int i = 1; i <= n; ++i) {
    cin >> str[i], len += (int)str[i].size();
    tr_insert(str[i]);
  }

  LL res = 2LL*len*n;
  for (int i = 1; i <= n; ++i) {
    string s = str[i]; reverse(s.begin(), s.end());
    res -= tr_query(s)*2;
  }
  cout << res << endl;
  return 0;
}
```

CF459E Pashmak and Graph

把所有边按照权值进行分类, 从小到大遍历每种权值的边, 进行 dp

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

using namespace std;

const int maxn = 3e5 + 5;
struct node {
  int u, v;
};
vector<node> vec[maxn];
int p[maxn], f[maxn];

int main()
{
  int n, m; cin >> n >> m;
  while (m -- ) {
    int u, v, w; cin >> u >> v >> w; vec[w].push_back({u,v});
  }

  for (int i = 1; i <= 100000; ++i) {
    for (node it : vec[i]) f[it.v] = max(f[it.v], p[it.u]+1);
    for (node it : vec[i]) p[it.v] = f[it.v];
  }

  int res = 0;
  for (int i = 1; i <= n; ++i) res = max(res, f[i]);
  cout << res << endl;
```

```
  return 0;
}
```

CF597C Subsequences

f[i][j]: 上升子序列的长度为 i, 以 j 这个值结尾的方案数有多少

f[i][j] <- f[i-1][1] + f[i-1][2] + f[i-1][3] + ... + f[i-1][j-1] (可以用树状数组快速求和)

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

using namespace std;

typedef long long LL;
const int N = 1e5 + 5, M = 12;
LL tr[M][N];

int lowbit(int x) { return x&(-x); }
void update(int id, int x, LL k) {
  while (x < N) { tr[id][x] += k; x += lowbit(x); }
}
LL query(int id, int x) {
  LL res = 0;
  while (x) { res += tr[id][x]; x -= lowbit(x); }
  return res;
}

int main()
{
  int n, k; cin >> n >> k;
  LL res = 0;
  update(0, 1, 1);
  while (n -- ) {
    int x; cin >> x; ++x;
    for (int i = k+1; i >= 1; --i) {
      LL t = query(i-1, x-1);
      update(i, x, t);
      if (i == k+1) res += t;
    }
  }
  cout << res << endl;
  return 0;
}
```