

Luyện tập C++: Set và Unordered Set

Danh sách bài tập Codeforces Elo 800 - 1000

Slide Learning CPP

Ngày 15 tháng 1 năm 2026

Mục tiêu

Việc luyện tập theo nhóm chủ đề giúp tạo ra các "chunk" kiến thức vững chắc. Danh sách này tập trung vào set và unordered_set.

- `std::set`: Tự động sắp xếp, phần tử duy nhất.
- `std::unordered_set`: Không sắp xếp, phần tử duy nhất, tốc độ nhanh hơn.

1. Boy or Girl (236A)

Link: Codeforces 236A

Tóm tắt đề bài

Đếm số ký tự khác nhau trong tên người dùng.

Chiến lược giải quyết

Đưa các ký tự của chuỗi vào một `set<char>`. Sau đó kiểm tra `set.size()` là số chẵn hay số lẻ để đưa ra kết luận.

2. Is your horseshoe on the other hoof? (228A)

Link: Codeforces 228A

Tóm tắt đề bài

Cho 4 màu móng ngựa, cần mua thêm ít nhất bao nhiêu móng để có đủ 4 màu khác nhau?

Chiến lược giải quyết

Đưa 4 giá trị màu vào set. Số lượng móng cần mua thêm sẽ bằng $4 - \text{set.size()}$.

3. Pangram (520A)

Link: Codeforces 520A

Tóm tắt đề bài

Kiểm tra một chuỗi có chứa đủ 26 chữ cái tiếng Anh (không phân biệt hoa thường) hay không.

Chiến lược giải quyết

Chuyển tất cả ký tự về chữ thường (lowercase), bỏ vào một `set<char>`. Kiểm tra xem `set.size()` có bằng 26 hay không.

4. Registration System (4C)

Link: Codeforces 4C

Tóm tắt đề bài

Quản lý tên người dùng. Nếu tên đã tồn tại, thêm số thứ tự tăng dần vào sau tên.

Chiến lược giải quyết

Sử dụng `map<string, int>` để lưu trữ và đếm số lần xuất hiện của mỗi tên người dùng. Đây là bài tập chuyển tiếp từ set sang map.

5. Anton and Letters (443A)

Link: Codeforces 443A

Tóm tắt đề bài

Đếm các chữ cái khác nhau xuất hiện trong một tập hợp có định dạng $\{a, b, c\}$.

Chiến lược giải quyết

Xử lý chuỗi đầu vào để lọc lấy các ký tự chữ cái, bỏ chúng vào set rồi in ra `set.size()`.

6. I Wanna Be the Guy (469A)

Link: Codeforces 469A

Tóm tắt đề bài

Hai người chơi có các danh sách màn chơi họ có thể vượt qua. Liệu cả hai có thể cùng nhau vượt qua tất cả n màn chơi không?

Chiến lược giải quyết

Thực hiện phép hợp (Union) bằng cách bỏ tất cả các màn chơi của cả hai người vào một set. Kiểm tra `set.size() == n`.

7. Arrival of the General (144A)

Link: Codeforces 144A

Tóm tắt đề bài

Tìm số bước hoán đổi ít nhất để đưa phần tử lớn nhất về đầu và phần tử nhỏ nhất về cuối.

Chiến lược giải quyết

Luyện tập tìm vị trí Max/Min. set có thể dùng để lưu các giá trị duy nhất nếu bài toán yêu cầu mở rộng về sau.

8. Gamer Hemose (1592A)

Link: Codeforces 1592A

Tóm tắt đề bài

Chọn 2 vũ khí mạnh nhất để tiêu diệt quái vật sao cho không dùng một vũ khí 2 lần liên tiếp.

Chiến lược giải quyết

Dùng set (để tự động sắp xếp) hoặc sort để lấy ra 2 giá trị sát thương lớn nhất.

9. Double Strings (1703D)

Link: Codeforces 1703D

Tóm tắt đề bài

Kiểm tra một chuỗi có bằng tổng (ghép) của 2 chuỗi khác có trong danh sách không.

Chiến lược giải quyết

Bỏ tất cả các chuỗi vào `unordered_set`. Với mỗi chuỗi, dùng vòng lặp cắt làm đôi tại mọi vị trí khả thi và dùng `count()` để kiểm tra sự tồn tại của cả hai nửa trong set.

10. Favorite Sequence (1462A)

Link: Codeforces 1462A

Tóm tắt đề bài

Khôi phục dãy số gốc từ một dãy bị xáo trộn theo quy tắc lấy luân phiên từ hai đầu.

Chiến lược giải quyết

Sử dụng cấu trúc dữ liệu mảng hoặc deque. Có thể dùng set để đánh dấu các phần tử đã xử lý nếu cần.

11. Two-gram (977B)

Link: Codeforces 977B

Tóm tắt đề bài

Tìm cặp 2 ký tự (substring độ dài 2) xuất hiện nhiều nhất trong chuỗi.

Chiến lược giải quyết

Dùng `map<string, int>` để đếm số lần xuất hiện của tất cả các cặp con độ dài 2.

12. Polycarp and Letters (864B)

Link: Codeforces 864B

Tóm tắt đề bài

Tìm đoạn con dài nhất chỉ chứa các chữ cái thường khác nhau.

Chiến lược giải quyết

Duyệt chuỗi: nếu gặp chữ hoa thì `clear()` set, nếu gặp chữ thường thì `insert()` vào set. Cập nhật kết quả cực đại từ `set.size()`.

16. Distinct Digits (1228A)

Link: Codeforces 1228A

Tóm tắt đề bài

Tìm một số trong đoạn $[l, r]$ mà tất cả các chữ số của nó đều khác nhau.

Chiến lược giải quyết

Duyệt từng số, chuyển số thành string, bỏ từng ký tự số vào `set<char>`. Nếu `set.size() == string.length()` thì đó là số cần tìm.

Lời khuyên khi giải bài

- **Tại sao lại có Map?** set để kiểm tra "có tồn tại hay không", còn map để biết "tồn tại bao nhiêu lần".
- **Khi nào dùng Unordered?** Với Elo 800-1000 ($n \leq 10^5$), `std::set` vẫn chạy rất tốt. Hãy tập dùng `std::set` trước để làm quen với tính chất sắp xếp.

Ví dụ: Boy or Girl (236A)

```
1 #include <iostream>
2 #include <set>
3 #include <string>
4
5 int main() {
6     std::string s;
7     std::cin >> s;
8     std::set<char> distinct_chars;
9     for(char c : s) {
10         distinct_chars.insert(c);
11     }
12
13     if(distinct_chars.size() % 2 == 0)
14         std::cout << "CHAT WITH HER!";
15     else
16         std::cout << "IGNORE HIM!";
17     return 0;
18 }
```

Ví dụ: Boy or Girl (236A)

```
1 #include <iostream>
2 #include <set>
3 #include <string>
4
5 int main() {
6     std::string s;
7     std::cin >> s;
8     std::set<char> distinct_chars;
9     for(char c : s) {
10         distinct_chars.insert(c);
11     }
12
13     if(distinct_chars.size() % 2 == 0)
14         std::cout << "CHAT WITH HER!";
15     else
16         std::cout << "IGNORE HIM!";
17     return 0;
18 }
```

Kết quả

Chương trình sẽ đếm số ký tự duy nhất và đưa ra quyết định.