

# Phân Tích Đề Thi Olympic Tin Học & ICPC UMT 2025

Nguyễn Quân Bá Hồng\*

Ngày 10 tháng 7 năm 2025

## Tóm tắt nội dung

This text is a part of the series *Some Topics in Advanced STEM & Beyond*:

URL: [https://nqbh.github.io/advanced\\_STEM/](https://nqbh.github.io/advanced_STEM/).

Latest version:

- *Phân Tích Đề Thi Olympic Tin Học & ICPC UMT 2025*.

PDF: URL: [.pdf](#).

TEX: URL: [.tex](#).

- .

PDF: URL: [.pdf](#).

TEX: URL: [.tex](#).

## Mục lục

<b>1 Olympic Tin Học Sinh Viên UMT 2025</b>	<b>1</b>
<b>2 ICPC UMT 2025</b>	<b>2</b>
<b>3 Miscellaneous</b>	<b>2</b>

## 1 Olympic Tin Học Sinh Viên UMT 2025

### Resources – Tài nguyên.

1. LÊ PHÚC LỮ. *Đề Thi Chính Thức Olympic Tin Học Sinh Viên UMT 2025*.
2. PHAN VINH TIẾN, ĐỖ ANH KIẾT, ĐẶNG PHÚC AN KHANG, NGÔ HOÀNG TÙNG, NGUYỄN LÊ ĐĂNG KHOA, TRẦN QUANG SƠN. *Lời Giải Tham Khảo Olympic Tin Học Sinh Viên & ICPC UMT 2025*. Fanpage: <https://www.facebook.com/STAC.UMT>.

**Bài toán 1** (Team ICPC cho mùa giải mới). Cho  $n \in \mathbb{N}^*$  thành viên, mỗi người có điểm năng lực là số nguyên  $a_i$ ,  $\forall i \in [n]$ . Đếm số cách chọn 1 đội gồm 3 thành viên sao cho hiệu giữa điểm cao nhất & thấp nhất trong đội không vượt quá 2.

Input. Dòng 1 chứa số nguyên dương  $n \in [3, 10^5]$ . Dòng 2 chứa  $n$  số nguyên dương  $a_i \in [10^9]$ ,  $\forall i \in [n]$ .

Output. Tổng số cách chọn đội thỏa mãn yêu cầu đề bài.

Subtask. 50% số điểm tương ứng với  $n \leq 10$ . 50% số điểm không có ràng buộc gì thêm.

1st solution. Sử dụng vét cạn, duyệt qua các cách chọn đội có thể có.

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main() {
5     int n; cin >> n;
6     vector<int> a(n);
7     for (int i = 0; i < n; ++i) cin >> a[i];
8     sort(a.begin(), a.end());
9
10    int res = 0;
11    for (int i = 0; i < n - 2; ++i)
12        for (int j = i + 1; j < n - 1; ++j)
```

\*A scientist- & creative artist wannabe, a mathematics & computer science lecturer of Department of Artificial Intelligence & Data Science (AIDS), School of Technology (SOT), UMT Trường Đại học Quản lý & Công nghệ TP.HCM, Hồ Chí Minh City, Việt Nam.  
E-mail: [nguyenquanbahong@gmail.com](mailto:nguyenquanbahong@gmail.com) & [hong.nguyenquanba@umt.edu.vn](mailto:hong.nguyenquanba@umt.edu.vn). Website: <https://nqbh.github.io/>. GitHub: <https://github.com/NQBH>.

```

13         for (int k = j + 1; k < n; ++k)
14             if (a[k] - a[i] <= 2) ++res;
15     cout << res;
16 }

```

*Phân tích độ phức tạp thuật toán:* Về độ phức tạp không gian, tốn  $n + 5 = O(n)$  ô nhớ `int` cho vector  $\{a[i]\}_{i=0}^{n-1}$ , & 5 biến  $n, i, j, k, \text{res}$ . Về độ phức tạp thời gian, tốn  $n + 1$  để nhập  $n$  & vector  $\{a[i]\}_{i=0}^{n-1}$ , `sort` tốn trung bình  $O(n \log n)$  time.  $\square$

## 2 ICPC UMT 2025

**Problem 1** (Smart Blueprint). *As part of the “Smart Campus” initiative at UMT University, a senior student team is digitizing every engineering blueprint of the underground infrastructure. After scanning the legacy drawings, they recovered a large collection of straight-line segments – each one representing a stretch of pipe, buried cable, or power line. Because different construction crews often redrew the same objects, many of these segments lie on the same infinite line & overlap one another wholly or partially.*

*To simplify the blueprint, the team must keep the minimum number of distinct segments that still preserves all information. 2 segments are said to overlap if they lie on the same straight line & have an intersection of positive length.*

**Input.** *The input consists of multiple test cases. Each test case begins with an integer  $n \in [10^4]$  – the number of line segments in the map. Then follow  $n$  lines, each containing 4 real numbers  $x_1, y_1, x_2, y_2 \in \mathbb{R}$  which represent the 2 endpoints of a line segment. Coordinates are in the range  $[0, 1000]$ , with up to 2 decimal places. No segment has zero length. The input ends with a line containing a single 0 – this line should not be processed.*

**Output.** *For every test case, output a single line with 1 integer – the minimum number of distinct segments that must remain.*

**Sample.**

smart_blueprint.inp	smart_blueprint.out
3	2
1.0 10.0 3.0 14.0	1
0.0 0.0 20.0 20.0	2
10.0 28.0 2.0 12.0	
2	
0.0 0.0 1.0 1.0	
1.0 1.0 2.15 2.15	
2	
0.0 0.0 1.0 1.0	
1.0 1.0 2.15 2.16	
0	

**Bài toán 2.** *Là một phần của sáng kiến “Smart Campus” tại Đại học UMT, một nhóm sinh viên năm cuối đang số hóa mọi bản thiết kế kỹ thuật của cơ sở hạ tầng ngầm. Sau khi quét các bản vẽ cũ, họ đã khôi phục được một bộ sưu tập lớn các đoạn thẳng – mỗi đoạn đại diện cho một đoạn ống, cáp ngầm hoặc đường dây điện. Vì các đội thi công khác nhau thường vẽ lại cùng một đối tượng, nên nhiều đoạn trong số này nằm trên cùng một đường thẳng vô hạn & chồng lên nhau toàn bộ hoặc một phần.*

*Để đơn giản hóa bản thiết kế, nhóm phải giữ số lượng đoạn riêng biệt tối thiểu vẫn bảo toàn được mọi thông tin. 2 đoạn được gọi là chồng lên nhau nếu chúng nằm trên cùng một đường thẳng & có giao điểm có độ dài dương.*

**Đầu vào.** *Đầu vào bao gồm nhiều trường hợp thử nghiệm. Mỗi trường hợp thử nghiệm bắt đầu bằng một số nguyên  $n \in [10^4]$  – số đoạn thẳng trong bản đồ. Sau đó, theo dõi  $n$  dòng, mỗi dòng chứa 4 số thực  $x_1, y_1, x_2, y_2 \in \mathbb{R}$  biểu diễn 2 điểm cuối của một đoạn thẳng. Tọa độ nằm trong phạm vi  $[0, 1000]$ , với tối đa 2 chữ số thập phân. Không có đoạn nào có độ dài bằng không. Đầu vào kết thúc bằng một dòng chứa một số 0 – dòng này không được xử lý.*

**Đầu ra.** *Đối với mỗi trường hợp kiểm tra, đầu ra một dòng duy nhất với 1 số nguyên – số lượng đoạn thẳng riêng biệt tối thiểu phải còn lại.*

## 3 Miscellaneous