



# BAN CHUYÊN MÔN

*Kế hoạch hoạt động*

*thành viên Ban Chuyên môn Câu lạc bộ Lập trình ứng dụng (APC)*

Câu lạc bộ Lập trình ứng dụng - Applied Programming Club

Câu lạc bộ trực thuộc UMT - Khoa Công nghệ

# Mục lục

1 KẾ HOẠCH HOẠT ĐỘNG DỰ KIẾN NĂM 2025 – 2026	2
2 LỘ TRÌNH HỌC C++ - COMPETITIVE PROGRAMMING	4

# CHƯƠNG 1

## KẾ HOẠCH HOẠT ĐỘNG DỰ KIẾN NĂM 2025 – 2026

### Hoạt động 1: Chương trình OLP Coaching

- **Thời gian triển khai:** Bắt đầu từ 21/12/2025 – kết thúc trước Kỳ thi OLP Tin học Sinh viên UMT lần 3 (2026) khoảng 02 tuần.
- **Hình thức tổ chức:** Online/Offline – tần suất 01-02 buổi/tuần.
- **Mục tiêu:**
  - Trang bị kiến thức lập trình căn bản đến trung cấp bằng ngôn ngữ C++.
  - Hình thành tư duy thuật toán và kỹ năng giải quyết bài toán lập trình.
  - Hỗ trợ thành viên có khả năng tham gia và làm bài thi OLP Tin học Sinh viên UMT.
- **Đối tượng tham gia:** Thành viên Ban Chuyên môn và các thành viên ban khác có nhu cầu.
- **Nội dung triển khai:**
  - *Học C++ và Thuật toán cơ bản:*
    - \* Biên soạn giáo trình nội bộ, tổ chức bài giảng.
    - \* Thực hành qua hệ thống bài tập Codeforces/VNOI/MarisaOJ/UMTOJ.
  - *Hướng dẫn sử dụng L<sup>A</sup>T<sub>E</sub>X:*
    - \* Viết báo cáo tổng hợp kiến thức sau mỗi giai đoạn học.

### Hoạt động 2: Chương trình OLP Mentoring – Đội tuyển Olympic Tin học

- **Thời gian triển khai:** Bắt đầu sau khi kết thúc Kỳ thi OLP Tin học Sinh viên UMT lần 3 và kết thúc 01 tuần trước Kỳ thi OLP Tin học Sinh viên Toàn quốc 2026.
- **Hình thức tổ chức:** Online/Offline – tần suất 01 buổi/tuần.
- **Mục tiêu:**
  - Nâng cao kiến thức về Cấu trúc dữ liệu & Thuật toán cho đội tuyển.
  - Rèn luyện chiến thuật thi đấu, tối ưu thời gian làm bài.
  - Tăng khả năng cạnh tranh tại Kỳ thi OLP Tin học Sinh viên Toàn quốc 2026.
- **Yêu cầu ràng buộc:**
  - Ít nhất 01 thành viên APC đạt suất thi OLP Tin học Quốc gia.
- **Đối tượng tham gia:**
  - Thành viên đội tuyển OLP Tin học thuộc CLB APC.
  - Các thành viên khác có năng lực phù hợp và mong muốn rèn luyện.

## **Hoạt động 3: APC Contest – Cuộc thi lập trình nội bộ**

- **Thời gian tổ chức:** Triển khai định kỳ trong học kỳ Spring & Summer.
- **Mục tiêu:**
  - Tạo môi trường thi đấu lập trình mô phỏng Olympic Tin học (IOI format).
  - Đào tạo đội ngũ *Problem Setter*, *Tester* và *Contest Organizer* nội bộ.
- **Nội dung triển khai:**
  - *Xây dựng đề thi:*
    - \* Hướng dẫn phân tích đề, chia *subtasks*, đánh giá độ khó.
  - *Sinh test & chấm tự động:*
    - \* Hướng dẫn viết test generator, validator, script chấm tự động.
  - *Quản lý hệ thống cuộc thi:*
    - \* Sử dụng GitHub và nền tảng UMTOJ để tổ chức thử nghiệm.

## **Hoạt động 4: Hỗ trợ Khoa Công nghệ tổ chức các kỳ thi/Workshop**

- (Nội dung sẽ bổ sung sau – theo kế hoạch của Khoa Công nghệ UMT)

# CHƯƠNG 2

## LỘ TRÌNH HỌC C++ - COMPETITIVE PROGRAMMING

### Phần I: C++ Cơ bản

1. Kiểu dữ liệu cơ bản C++ (nguyên, thực, chuỗi, ...) và câu lệnh **if**
2. Phạm vi của biến (Enclosing Scope, Local Scope, Global Scope)
3. Vòng lặp (**for**, **while**) và độ phức tạp thuật toán
4. Mảng 1 chiều, **vector**, mảng 2 chiều  
Thuật toán sắp xếp & tìm kiếm
5. Chuỗi ký tự (**string**)
6. Hàm (Chương trình con)
7. Dê quy

### Phần II: Cấu trúc dữ liệu & Thuật toán cơ bản

#### 1. Cấu trúc dữ liệu cơ bản

[https://github.com/GrootTheDeveloper/OLP-ICPC/blob/master/Algo\\_Notes/DataStructure/main.pdf](https://github.com/GrootTheDeveloper/OLP-ICPC/blob/master/Algo_Notes/DataStructure/main.pdf)

- a) Ngăn xếp (**stack**)
- b) Hàng đợi (**queue**), hàng đợi 2 đầu (**dequeue**)
- c) Hàng đợi ưu tiên (**priority\_queue** / **heap**)
- d) Cây tìm kiếm nhị phân: **set**, **map**

#### 2. Kỹ thuật lập trình

[https://github.com/GrootTheDeveloper/OLP-ICPC/blob/master/Algo\\_Notes/Techniques/main.pdf](https://github.com/GrootTheDeveloper/OLP-ICPC/blob/master/Algo_Notes/Techniques/main.pdf)

- a) Hai con trỏ (Two Pointers)
- b) Tìm kiếm nhị phân trên kết quả
- c) Tìm kiếm nhị phân trên số thực
- d) Tham lam (Greedy)
- e) Dê quy & quay lui (Backtracking), chia để trị (Divide & Conquer)

#### 3. Quy hoạch động (Dynamic Programming)

[https://github.com/GrootTheDeveloper/OLP-ICPC/blob/master/Algo\\_Notes/DynamicProgramming/main.pdf](https://github.com/GrootTheDeveloper/OLP-ICPC/blob/master/Algo_Notes/DynamicProgramming/main.pdf)

- a) Bài toán QHD có điển
- b) QHD không có điển
- c) Kỹ thuật đổi biến số trong QHD

#### **4. Lý thuyết đồ thị (Graph Theory)**

[https://github.com/GrootTheDeveloper/OLP-ICPC/blob/master/Algo\\_Notes/Graph/main.pdf](https://github.com/GrootTheDeveloper/OLP-ICPC/blob/master/Algo_Notes/Graph/main.pdf)

- a) Giới thiệu về đồ thị & thuật toán DFS
- b) Thuật toán BFS
- c) Cây DFS và ứng dụng (khớp, cầu, SCC)
- d) Disjoint Set Union (DSU)

#### **5. Toán học**

[https://github.com/GrootTheDeveloper/OLP-ICPC/blob/master/Algo\\_Notes/Mathematics/main.pdf](https://github.com/GrootTheDeveloper/OLP-ICPC/blob/master/Algo_Notes/Mathematics/main.pdf)

- a) Tổ hợp
- b) Quy hoạch động + Tổ hợp
- c) Xác suất
- d) Kỳ vọng
- e) Lý thuyết trò chơi (Game Theory)

### **Phần III: Cấu trúc dữ liệu & Thuật toán nâng cao**

#### **1. Cấu trúc dữ liệu nâng cao**

- a) Segment Tree
- b) Lazy Propagation
- c) Persistent Segment Tree
- d) Fenwick Tree (2D/3D)
- e) Trie
- f) Sparse Table
- g) Suffix Array

#### **2. Kỹ thuật lập trình**

- a) Merge Sort Tree
- b) Bit Manipulation
- c) Hashing
- d) Sweep Line
- e) Random Techniques
- f) Meet in The Middle

#### **3. Quy hoạch động**

- a) QHD Bitmask
- b) Sum Over Subset (SOS)
- c) QHD chữ số
- d) QHD nhân ma trận
- e) QHD trên cây

#### **4. Lý thuyết đồ thị nâng cao**

- a) Maximum Flow & Matching
- b) Minimum Cut
- c) Lowest Common Ancestor (LCA)
- d) Heavy-Light Decomposition (HLD)
- e) Centroid Decomposition

#### **5. Toán học nâng cao**

- a) Bao hàm – Loại trừ (Inclusion–Exclusion)
- b) Grundy Number
- c) Euler Totient Function
- d) Möbius Function
- e) Hình học (Geometry)
- f) Convex Hull & Tối ưu QHD
- g) FFT

