

ขอบเขตเนื้อหาวิชาที่ครอบคลุมในการแข่งขันคอมพิวเตอร์โอลิมปิกระดับชาติ

ภาพรวม

ขอบเขตเนื้อหาวิชาที่อยู่ในการแข่งขันคอมพิวเตอร์โอลิมปิกระดับชาติคล้ายกับขอบเขตเนื้อหาวิชาโครงสร้างข้อมูลและอัลกอริทึมที่มีการสอนในหลักสูตรปริญญาตรี ซึ่งจุดนี้ก็คล้ายกับขอบเขตเนื้อหาวิชาในการแข่งขันระดับนานาชาติด้วย ประเด็นที่น่าสนใจมีสองประเด็นดังนี้

1. ขอบเขตเนื้อหาที่ไม่ครอบคลุมในการแข่งขันระดับชาติ แต่ครอบคลุมในการแข่งขันระดับนานาชาติ ได้แก่ เนื้อหาในเรื่องอัลกอริทึมเรขาคณิต (Geometric Algorithms) เช่น คุณสมบัติของส่วนของเส้นตรง ตำแหน่งจุดที่เกี่ยวข้องกับโพลิกอน และการหาเปลือกหุ้มแบบคอนเวกซ์ (Convex Hull) เป็นต้น เนื้อหาเรื่องอัลกอริทึมการประมาณแบบไม่ต่อเนื่อง (Discrete Approximation Algorithm) และต้นไม้เฟนวิก (Fenwick Trees)
2. ขอบเขตเนื้อหาที่ไม่ครอบคลุมทั้งในการแข่งขันระดับชาติและการแข่งขันระดับนานาชาติ ได้แก่ เนื้อหาเรื่อง วิทยาการศึกษาลำบาก (Heuristics) อัลกอริทึมการประมาณเชิงตัวเลข (Numerical Approximation Algorithm) อัลกอริทึมการไหลสูงสุด (Maximum Flow Algorithms) อัลกอริทึมการจับคู่แบบสองส่วน (Bipartite Matching Algorithms) ส่วนประกอบที่เชื่อมกันแบบเข้มข้นในกราฟระบุทิศทาง (Strongly Connected Component in Directed Graph) (แต่ส่วนประกอบที่เชื่อมกัน (Connected Component) และส่วนประกอบที่เชื่อมกันแบบสองเส้นทาง (Biconnected Component) ในกราฟแบบไม่มีทิศทางอยู่ในขอบเขตเนื้อหาของการแข่งขันระดับชาติ), อัลกอริทึมเชิงขนาน (Parallel Algorithms) และโครงสร้างข้อมูลฮีบแบบทวิภาคซับซ้อน (Complex Binary Heap Data Structures) เช่น ฮีบแบบทวินาม (Binomial Heaps) และ ฮีบแบบฟีโบนัชชี (Fibonacci Heaps) เป็นต้น (แต่ฮีบแบบทวิภาค (Binary Heap) พื้นฐานอยู่ในขอบเขตเนื้อหาการแข่งขันระดับชาติ)

ขอบเขตเนื้อหาวิชา

แบ่งได้เป็น 3 หมวดคือ (1) คณิตศาสตร์ (2) พื้นฐานวิทยาการคอมพิวเตอร์ และ (3) อัลกอริทึม

1. หมวดคณิตศาสตร์

1.1 เลขคณิตและเรขาคณิต

- 1.1.1 จำนวนเต็ม คุณสมบัติของเลขจำนวนเต็ม (ค่าบวก ค่าลบ เลขคู่ เลขคี่ การหารลงตัวจำนวนเฉพาะ)
- 1.1.2 เลขเศษส่วน และร้อยละ
- 1.1.3 จุด เวกเตอร์ พิกัดจุดแบบคาร์ทีเซียน (Cartesian Coordinates) ในตารางสองมิติที่มีพิกัดเป็นจำนวนเต็ม
- 1.1.4 ระยะทางแบบยูคลิด ทฤษฎีพีทาโกรัส
- 1.1.5 ส่วนของเส้นตรง จุดตัดของเส้นตรง และสมบัติพื้นฐานที่เกี่ยวข้อง
- 1.1.6 มุม สามเหลี่ยม สี่เหลี่ยมผืนผ้า สี่เหลี่ยมจัตุรัส วงกลม

1.2 โครงสร้างไม่ต่อเนื่อง (Discrete Structures)

1.2.1 ฟังก์ชัน ความสัมพันธ์ และเซต

1.2.2 ตรรกศาสตร์พื้นฐาน

1.2.3 วิธีการพิสูจน์

1.2.4 วิธีการนับเบื้องต้น

1.2.4.1 กฎการหาผลรวมของผลคูณ (Sum of Product Rule) หลักการเพิ่มเข้า-ตัดออก (Inclusion-exclusion Principle) ลำดับเลขคณิตและเรขาคณิต จำนวนแบบฟีโบนัชชี (Fibonacci Numbers)

1.2.4.2 กฎรังนกพิราบ (Pigeonhole Principle) เพื่อใช้ในการหาขอบเขต

1.2.4.3 การเรียงสับเปลี่ยน และวิธีจัดหมู่ระดับพื้นฐาน

1.2.4.4 ฟังก์ชันเลขเศษส่วน (Fractional Function) และสัมประสิทธิ์ทวินาม (Binomial Coefficient)

1.2.5 กราฟและต้นไม้

1.2.5.1 ต้นไม้และคุณสมบัติพื้นฐาน

1.2.5.2 กราฟไม่มีทิศทาง (Degree, Path, Cycle, Connectedness)

1.2.5.3 กราฟแบบมีทิศทาง (In-degree, Out-degree, Directed path/cycle)

1.2.5.4 Spanning Trees

1.2.5.5 วิธีการเดินผ่านต้นไม้ (Traversal Strategies: Defining the Node Order for Ordered Trees)

1.2.5.6 'Decorated' Graphs with Edge/Node Labels, Weights, Colors

1.2.5.7 Multigraphs และ Graphs ที่มี Self-Loops

หมายเหตุ การแข่งขันไม่ครอบคลุมเนื้อหาเรื่อง Planar Graphs, Bipartite Graphs และ Hypergraphs

1.3 เนื้อหาที่ไม่รวมอยู่ในการแข่งขัน

1.3.1 แคลคูลัส

1.3.2 ความน่าจะเป็น

1.3.3 สถิติ

1.3.4 จำนวนจริงและจำนวนเชิงซ้อน

1.3.5 ภาคตัดกรวยทั่วไป (Parabolas, Hyperbolas, Ellipses) แต่เรื่องวงกลมอยู่ภายใต้ขอบเขตเนื้อหาในการแข่งขัน

ระดับชาติ

1.3.6 โพลิกอน (ในระดับนานาชาติจะครอบคลุมเนื้อหาเกี่ยวกับโพลิกอน)

2. หมวดพื้นฐานวิทยาการคอมพิวเตอร์

2.1 พื้นฐานด้านการเขียนโปรแกรม

2.2 ทักษะการแก้ปัญหา (Problem-solving Skill)

2.3 พื้นฐานโครงสร้างข้อมูล

2.3.1 ชนิดข้อมูลดั้งเดิม (Primitive Data Type) ได้แก่ Boolean, Signed/Unsigned Integer, Character

2.3.2 แถวลำดับ (อาร์เรย์ อาร์เรย์หลายมิติ)

2.3.3 Record/Struct

2.3.4 สตริงและการดำเนินการกับสตริง

2.3.5 Static และ Stack Allocation

2.3.6 Lined Structures (ทั้งที่เป็นแบบเส้นตรง และแบบที่แบ่งเป็นสาขาได้)

2.3.7 การสร้าง โครงสร้างกองซ้อน (Stack), คิว (Queue), ต้นไม้ และกราฟ

2.3.8 การเลือกโครงสร้างข้อมูลที่เหมาะสม

2.3.9 คิวลำดับความสำคัญ (Priority Queue), ไดนามิกเซต (Dynamic Set), ไดนามิกแมป (Dynamic Map)

2.4 การเรียกตัวเองซ้ำ (Recursion)

2.4.1 แนวคิด

2.4.2 ฟังก์ชันทางคณิตศาสตร์ที่เรียกตัวเองซ้ำ

2.4.3 วิธีแบ่งแยกและเอาชนะ (Divide and Conquer)

2.4.4 อัลกอริทึมการย้อนรอยแบบเรียกตัวเองซ้ำ (Recursive Backtracking)

3. หมวดอัลกอริทึม

3.1 พื้นฐานการวิเคราะห์ความซับซ้อนของอัลกอริทึม (Algorithmic Complexity)

3.2 กลวิธีทางอัลกอริทึม

- 3.2.1 Brute-Force Algorithm
- 3.2.2 Greedy Algorithm
- 3.2.3 การแบ่งแยกและเอาชนะ
- 3.2.4 Backtracking (ทั้งที่เป็นแบบเรียกตัวเองซ้ำ และไม่เรียกตัวเองซ้ำ)
- 3.2.5 Branch-and-Bound Algorithm
- 3.2.6 Pattern matching and String/Text Algorithm
- 3.2.7 Dynamic Programming
- 3.3 อัลกอริทึมเชิงคำนวณพื้นฐาน
 - 3.3.1 อัลกอริทึมเชิงตัวเลขพื้นฐานที่เกี่ยวข้องกับจำนวนเต็ม เช่น
 Radix Conversion, Euclid's algorithm, Primality test in $O(N^{1/2})$,
 Sieve of Eratosthenes, Factorization, Efficient Exponentiation
 - 3.3.2 การจัดการอาร์เรย์ขั้นพื้นฐาน (รวมถึงการทำฮีโธแกรม และ Bucket Sort)
 - 3.3.3 Sequential และ Binary search
 - 3.3.4 Search by Elimination
 - 3.3.5 การแบ่งข้อมูล (Partitioning) การจัดลำดับด้วยการแบ่งข้อมูลซ้ำๆ Quick Sort
 - 3.3.6 การเรียงข้อมูลที่มีเวลาที่แย่ที่สุดเป็น $O(N \log N)$ เช่น Heap Sort และ Merge Sort
 - 3.3.7 Binary Heap พื้นฐาน และ Binary Search Tree
 - 3.3.8 การบรรยายโครงสร้างกราฟ เช่น Adjacency List และ Adjacency Matrix
 - 3.3.9 Depth-First and Breadth-First Traversals of Graphs และการหาองค์ประกอบที่เชื่อมต่อกันของกราฟแบบไม่มีทิศทาง
 - 3.3.10 Shortest Path Algorithm เช่น Dijkstra, Bellman-Ford และ Floyd-Warshall
 - 3.3.11 Transitive Closure (Floyd's Algorithm)
 - 3.3.12 Minimum Spanning Tree
 - 3.3.13 Topological Sort