

ขอบเขตเนื้อหาวิชาที่ครอบคลุมในการแข่งขันคอมพิวเตอร์โอลิมปิกระดับชาติ

ภาพรวม

ขอบเขตเนื้อหาวิชาที่อยู่ในการแข่งขันคอมพิวเตอร์โอลิมปิกระดับชาติคล้ายกับขอบเขตเนื้อหาวิชาโครงสร้างข้อมูลและอัลกอริทึมที่มีการสอนในหลักสูตรปริญญาตรี ซึ่งจุดนี้ก็คล้ายกับขอบเขตเนื้อหาวิชาในการแข่งขันระดับนานาชาติด้วย ประเด็นที่น่าสนใจมีสองประเด็นดังนี้

1. ขอบเขตเนื้อหาที่ไม่ครอบคลุมในการแข่งขันระดับชาติ แต่ครอบคลุมในการแข่งขันระดับนานาชาติ ได้แก่ เนื้อหาในเรื่องอัลกอริทึมเรขาคณิต (Geometric Algorithms) เช่น คุณสมบัติของส่วนของเส้นตรง, ตำแหน่งจุดที่เกี่ยวข้องกับโพลี곤, และการหาเปลือกหุ้มแบบคอนเวกซ์ (Convex Hull) เป็นต้น เนื้อหาเรื่องอัลกอริทึมการประมาณแบบไม่ต่อเนื่อง (Discrete Approximation Algorithm) และ ต้นไม้เฟนวิก (Fenwick Trees)
2. ขอบเขตเนื้อหาที่ไม่ครอบคลุมทั้งในการแข่งขันระดับชาติและการแข่งขันระดับนานาชาติ ได้แก่ เนื้อหาเรื่อง วิทยาการศึกษานานุกรม (Heuristics), อัลกอริทึมการประมาณเชิงตัวเลข (Numerical Approximation Algorithm), อัลกอริทึมการไหลสูงสุด (Maximum Flow Algorithms), อัลกอริทึมการจับคู่แบบสองส่วน (Bipartite Matching Algorithms), ส่วนประกอบที่เชื่อมกันแบบเข้มข้นในกราฟระบุทิศทาง (Strongly Connected Component in Directed Graph) (แต่ ส่วนประกอบที่เชื่อมกัน (Connected Component) และส่วนประกอบที่เชื่อมกันแบบเข้มข้น (Strongly Connected Component) ในกราฟแบบไม่มีทิศทางอยู่ในขอบเขตเนื้อหาของการแข่งขันระดับชาติ), อัลกอริทึมเชิงขนาน (Parallel Algorithms) และ โครงสร้างข้อมูลฮีบแบบทวิภาคซับซ้อน (Complex Binary Heap Data Structures) เช่น ฮีบแบบทวินาม (Binomial Heaps) และ ฮีบแบบฟีโบนัชชี (Fibonacci Heaps) เป็นต้น (แต่ ฮีบแบบทวิภาค (Binary Heap) พื้นฐานอยู่ในขอบเขตเนื้อหาการแข่งขันระดับชาติ)

ขอบเขตเนื้อหาวิชา

แบ่งได้เป็น 3 หมวด คือ (1) คณิตศาสตร์ (2) พื้นฐานวิทยาการคอมพิวเตอร์ และ (3) อัลกอริทึม

1. หมวดคณิตศาสตร์

1.1 เลขคณิตและเรขาคณิต

1.1.1 จำนวนเต็ม คุณสมบัติของเลขจำนวนเต็ม (ค่าบวก ค่าลบ เลขคู่ เลขคี่ การหารลงตัว จำนวนเฉพาะ)

1.1.2 เลขเศษส่วน และร้อยละ

1.1.3 จุด เวกเตอร์ พิกัดจุดแบบคาร์ทีเซียน (Cartesian coordinates) ในตารางสองมิติที่มีพิกัดเป็นจำนวนเต็ม

1.1.4 ระยะทางแบบยูคลิด ทฤษฎีพีทาโกรัส

1.1.5 ส่วนของเส้นตรง จุดตัดของเส้นตรง และคุณสมบัติพื้นฐานที่เกี่ยวข้อง

1.1.6 มุม สามเหลี่ยม สีเหลี่ยมผืนผ้า สีเหลี่ยมจัตุรัส 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 กฎของการบวกและกฎของการคูณ (Sum rule and 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, Handshaking Lemma)

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) แต่เรื่องวงกลมอยู่ภายใต้ขอบเขตเนื้อหาในการแข่งขันระดับชาติ
- 1.3.6 พอลิกอน (ในระดับนานาชาติจะครอบคลุมเนื้อหาเกี่ยวกับพอลิกอน)

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 (อาเรย์ อาเรย์หลายมิติ)

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

โปรแกรมที่ติดตั้งในเครื่องคอมพิวเตอร์ที่ใช้ในการแข่งขัน

เครื่องคอมพิวเตอร์ที่ใช้ในการแข่งขันใช้ระบบปฏิบัติการ Linux โดยมีโปรแกรมที่ติดตั้งดังนี้

1. ระบบปฏิบัติการ Ubuntu 32 บิต (12.04.4)
2. Web browsers: Firefox (27.0.1)
3. Editors: vim (7.3.429), kate (3.8.5), kwrite (4.8.5), kdevelop (4.3.1), emacs (24.3.1), gedit (3.4.1), nano (2.2.6), codeblocks (13.12 rev 9639), geany (0.21), cream (0.43), scite (3.3.9), leafpad (0.8.18.1)
4. Compiler: gcc (4.7.3), g++ (4.7.3)
5. Debugger: gdb (7.4)

หมายเหตุ: ตัวเลขในวงเล็บคือหมายเลขเวอร์ชันของโปรแกรม ทางคณะกรรมการอาจเปลี่ยนแปลงเวอร์ชันของโปรแกรมต่างๆ เป็นเวอร์ชันที่ใหม่ขึ้น หากเห็นสมควร

ข้อกำหนดของระบบปฏิบัติการและตัวแปลภาษาที่ใช้ในการแข่งขัน

1. โปรแกรมที่ผู้เข้าแข่งขันจัดทำในระหว่างการแข่งขัน กำหนดให้เขียนตามมาตรฐานของภาษา C หรือภาษา C++ ไม่อนุญาตให้เขียนโปรแกรมที่ทำงานใน Graphic Mode
2. ฟังก์ชันทั้งหมดในการเขียนโปรแกรม กำหนดให้ใช้ฟังก์ชันจากคลังมาตรฐานของภาษา C (The Standard C Library), conio.h (เฉพาะการทำงานบนระบบปฏิบัติการวินโดวส์) และ Standard Template Library (STL) เท่านั้น
 - a. ไม่อนุญาตให้ใช้ฟังก์ชันจัดการกับแฟ้มและอุปกรณ์โดยตรงที่กำหนดรูปแบบใช้งานในแฟ้ม (fcntl.h), (io.h) และ (iomanip.h)
 - b. ไม่อนุญาตให้โปรแกรมสร้างแฟ้มข้อมูลสำรองเพิ่มเติมระหว่างการทำงาน ห้ามอ่านหรือเขียนแฟ้มข้อมูลอื่นนอกเหนือจากที่โจทย์ระบุ
 - c. ไม่อนุญาตให้เรียกใช้โปรแกรมอื่นๆ (เช่น ผ่านทางฟังก์ชัน system) หรือเรียกใช้ system call นอกเหนือจากที่ใช้งานปกติ
 - d. ไม่อนุญาตให้ทำการคำนวณแบบมัลติโปรเซสซิง (multi-processing) เช่น ไม่อนุญาตให้โปรแกรมเรียกใช้ฟังก์ชันใน thread library ต่างๆ
3. โปรแกรมภาษา C ที่ผู้เข้าแข่งขันจัดทำในระหว่างการแข่งขัน กำหนดให้เขียนโปรแกรมที่ส่วนขยายเป็น .c สำหรับภาษา C++ ให้ใช้นามสกุล .cpp และต้องอยู่ในรูปแบบที่สามารถแปล (compile) ให้เป็นโปรแกรมที่สามารถทำงานได้โดยสมบูรณ์จากบรรทัดคำสั่ง (command line)

4. ใช้ GCC (GNU compiler collection) ในการตรวจโปรแกรมเพื่อให้คะแนน โดยใช้วิธีการแปลและให้ทำงานจากบรรทัดคำสั่งเท่านั้น โปรแกรมจะถูกสั่งให้ทำงานบนระบบปฏิบัติการและคอมไพเลอร์เดียวกันกับผู้เข้าแข่งขันเลือกใช้ ทั้งนี้เครื่องที่ใช้ในการตรวจสอบคำตอบของผู้เข้าแข่งขันจะเลือกระบบปฏิบัติการและคอมไพเลอร์โดยพิจารณาข้อมูลจากที่กำหนดไว้ที่ต้นไฟล์คำตอบของผู้เข้าแข่งขัน (รายละเอียดเพิ่มเติมอยู่ในหัวข้อ ‘ข้อมูลและรายละเอียดเพิ่มเติมเกี่ยวกับการแข่งขัน’)
5. คอมไพเลอร์ออปชัน (compiler option) ที่ใช้ในการแข่งขันจะทำการออปติไมซ์ (optimize) โปรแกรมโดยใช้ออปชัน -O2

วิธีการติดตั้งระบบและโปรแกรมที่ใช้ในการแข่งขัน

ตั้งแต่การแข่งขันคอมพิวเตอร์โอลิมปิกระดับชาติครั้งที่ 10 เป็นต้นไปได้มีการกำหนดให้ใช้ระบบปฏิบัติการ Linux ในการแข่งขันเท่านั้น ทางศูนย์เจ้าภาพจึงได้จัดเตรียมวิธีการสำหรับติดตั้งระบบและโปรแกรมที่ใช้ในการแข่งขันเพื่อที่จะให้อาจารย์ประจำศูนย์ นักเรียนผู้เข้าแข่งขัน และผู้ที่มีความสนใจสนใจได้ทดลองใช้ทำความคุ้นเคยกับระบบปฏิบัติการและโปรแกรมที่จะใช้ในการแข่งขันจริง โดยมีวิธีการติดตั้งดังนี้

สำหรับผู้ใช้งานระบบปฏิบัติการ Window และต้องการทดลองระบบผ่าน Virtual Machine สามารถกระทำได้โดย เข้าถึงเว็บ “graphics1.sci.ubu.ac.th/download/ubutoi10” แล้วเลือกดาวน์โหลดไฟล์ “UBUTOI10.tar.gz” จากนั้นทำการแยกไฟล์ (Extract File) ที่ดาวน์โหลดนี้จะเจอไฟล์ vmdk ที่จำลองระบบพร้อมทั้งโปรแกรมที่ใช้ในการแข่งขัน

สำหรับผู้ที่ต้องการทดลองระบบจากระบบปฏิบัติการ Linux โดยตรง จำเป็นต้องทำการติดตั้งระบบปฏิบัติการ Ubuntu 12.04.4 ก่อนหรือติดตั้ง Ubuntu 12.04 แล้วทำการอัปเดตก็ได้ จากนั้นให้เข้าถึงเว็บ “graphics1.sci.ubu.ac.th/download/ubutoi10” แล้วเลือกดาวน์โหลดไฟล์ “install-offlinepackges.sh” จากนั้นทำการรันไฟล์ที่ดาวน์โหลดมาโดยคำสั่ง “`sudo ./install-offlinepackges.sh`” ผ่านทางเทอร์มินัล รอจนรันเสร็จเรียบร้อยแล้วจะเป็นการเสร็จสิ้นการติดตั้งโปรแกรมที่ใช้ในการแข่งขันทั้งหมด