

FACULTY OF ENGINEERING
CHULALONGKORN UNIVERSITY
2110211 INTRODUCTION TO DATA STRUCTURE
Year II, First Semester, Final Examination, Dec 6, 2023, 08:30 – 11:30

ชื่อ-นามสกุล.....เลขประจำตัว.....ตอนเรียนที่.....เลขที่ใน CR58.....

หมายเหตุ

1. ข้อสอบมีทั้งหมด 3 ส่วน ส่วนที่หนึ่งคือข้อสอบ 5 ข้อ ในกระดาษคำถามคำตอบ 6 หน้าต่อไปนี้
2. ไม่อนุญาตให้นำตำราและเอกสารใดๆ เข้าในห้องสอบ
3. ไม่อนุญาตให้ใช้เครื่องคำนวณใดๆ
4. ห้ามการหยิบยื่นสิ่งใดๆ ทั้งสิ้น จากผู้สอบอื่นๆ เว้นแต่เจ้าหน้าที่ควบคุมการสอบจะหยิบยื่นให้
5. ห้ามนำส่วนใดส่วนหนึ่งของข้อสอบและสมุดคำตอบออกจากห้องสอบ
6. ผู้เข้าสอบสามารถออกจากห้องสอบได้ หลังจากผ่านการสอบไปแล้ว 45 นาที
7. เมื่อหมดเวลาสอบ ผู้เข้าสอบต้องหยุดการเขียนใดๆ ทั้งสิ้น
8. **นิสิตกระทำผิดเกี่ยวกับการสอบ ตามข้อบังคับจุฬาลงกรณ์มหาวิทยาลัย มีโทษ คือ พ้นสภาพการเป็นนิสิต หรือ ได้รับสัญลักษณ์ F ในรายวิชาที่กระทำผิด และอาจพิจารณาให้ถอนรายวิชาอื่นทั้งหมดที่ลงทะเบียนไว้ในภาคการศึกษานี้**

ห้ามนิสิตพกโทรศัพท์และอุปกรณ์สื่อสารไว้กับตัวระหว่างสอบ หากตรวจพบจะถือว่า
นิสิตกระทำผิดเกี่ยวกับการสอบ อาจต้องพ้นสภาพการเป็นนิสิต หรือ ให้ได้รับ F และ
อาจพิจารณาให้ถอนรายวิชาอื่นทั้งหมดที่ลงทะเบียนไว้ในภาคการศึกษานี้

* ร่วมรณรงค์การไม่กระทำผิดและไม่ทุจริตการสอบ *

ข้าพเจ้ายอมรับในข้อกำหนดที่กล่าวมานี้ ข้าพเจ้าเป็นผู้ทำข้อสอบนี้ด้วยตนเองโดยมิได้รับการช่วยเหลือ หรือให้ความช่วยเหลือ ในการทำข้อสอบนี้

ลงชื่อนิสิต.....

วันที่.....

- ใช้ดินสอเขียนคำตอบได้
- ให้เขียนเลขที่ในใบเซ็นชื่อเข้าสอบทุกหน้า
- หากพื้นที่สำหรับเขียนคำตอบไม่เพียงพอ ให้เขียนไว้ด้านหลังของหน้านั้น ห้ามเขียนข้ามไปหน้าอื่น และให้ระบุไว้ในพื้นที่สำหรับเขียนคำตอบว่า “มีต่อด้านหลัง”

- ข้อสอบมี 3 ส่วน ส่วนแรกให้เขียนคำตอบในกระดาษคำตอบเท่านั้น
- ส่วน 2 ให้ทำในระบบ grader เท่านั้น (โจทย์อยู่ในระบบ grader)
- ส่วน 3 สามารถเลือกได้ว่าจะทำในกระดาษหรือทำใน grader

Part I จงเติมค่าลงในช่องว่างที่กำหนดไว้

1. (5 คะแนน) จงวิเคราะห์เวลาการทำงานของฟังก์ชันข้างล่างนี้ (ในรูปของ Θ ของฟังก์ชัน n)

<pre>int method1(int n) { std::vector<int> v; for (int i=0; i< n; i++) { v.clear(); for (int j=0; j< n; j++) v.push_back(1); } }</pre>	method1 ใช้เวลา เป็น $\Theta(?)$
<pre>int method1(int n) { std::vector<int> v(n,0); for (int i=0; i< n/2; i++) v.erase(v.begin() + n/2 - i - 1); }</pre>	method2 ใช้เวลา เป็น $\Theta(?)$
<pre>int method3(int n) { cp::map_avl<int,int> m; for (int i=0; i< n; i++) m[i] = i; int sum = 0; for (int i=0; i< n/2; i++) for (auto &x : m) sum+=m[x.first]; return sum; }</pre>	method3 ใช้เวลา เป็น $\Theta(?)$
<pre>void method4(int n) { std::list<int> l; for (int i=0; i< n; i++) l.insert(l.begin(),0); for (int i=0; i< n; i++) l.erase(l.begin()); }</pre>	method4 ใช้เวลา เป็น $\Theta(?)$
<pre>int method5(int n) { cp::map_bst<int,int> m; for (int i=n; i > 0; i--) m[i] = i; }</pre>	method5 ใช้เวลา เป็น $\Theta(?)$

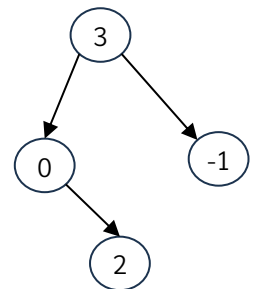
2. (5 คะแนน) จงวาดต้นไม้ AVL Tree ที่มี n ปม โดยที่แต่ละปมมีค่าอยู่ในช่วง 1 ถึง n โดยไม่ซ้ำกันเลย และ AVL Tree ต้นนี้ จะต้อง มีปมที่มีค่าเป็น a โดยที่ปมที่มีค่าเป็น $a+1$ นั้นจะต้องอยู่ในความลึกที่ต่างจากปมที่มีค่าเป็น a เท่ากับ 3 ชั้นพอดี โดย AVL Tree ต้นนี้ ต้องมีจำนวนปมน้อยที่สุดเท่าที่เป็นไปได้

3. (4 คะแนน) มี Binary Tree ต้นหนึ่ง (ไม่ใช่ Binary Search Tree) ต้นหนึ่งที่แต่ละปมเก็บข้อมูลจำนวนเต็ม หากเราทำการพิมพ์ข้อมูลของแต่ละปมในต้นไม้ตามลำดับ ตามลำดับ pre-order traversal จะได้ผลเป็น 1 2 4 3 5 7 8 6 9 10 แต่ถ้าหากพิมพ์ตามลำดับ in-order traversal จะได้ผลเป็น 4 2 1 7 5 8 3 10 9 6 จงวาดต้นไม้ต้นนี้

4. (4 คะแนน) จากส่วนของโปรแกรมต่อไปนี้ให้ class node คือปมของ Binary Tree จงวาด binary tree ที่เป็นผลลัพธ์ของการเรียกฟังก์ชัน recur(7) โดยวาดในรูปแบบตัวอย่างด้านขวามือนี้ (รูปด้านขวานี้ “อาจจะไม่ใช่” ผลลัพธ์ของการเรียกฟังก์ชัน recur(x) ก็ได้

```
class node {
public:
    int data;
    node *left, *right;
    node(int d,node* l, node* r) : data(d), left(l), right(r) {};
};
int count = 0;

node* recur(int d) {
    count++;
    if (d <= 0)
        return new node(d,NULL,NULL);
    return new node(d, recur(d/2), recur(d-2));
}
```



5. (20 คะแนน) ในข้อย่อยต่อไปนีให้ตอบโดยเลือกคำตอบที่ถูกต้องที่สุด แต่ละข้อย่อยมีคะแนน 1 คะแนน หากไม่ตอบในข้อใด จะได้คะแนน 0 แต่ถ้าหากตอบผิดในข้อใด จะได้คะแนน -0.5 ต่อข้อย่อย อย่างไรก็ตาม ถึงแม้จะตอบผิดจนได้คะแนนรวมติดลบ จะถือว่าข้อนี้ได้คะแนนเป็น 0 ****ให้เขียนคำตอบลงในตารางด้านล่างนี้เท่านั้น****

ข้อ 1	ข้อ 2	ข้อ 3	ข้อ 4	ข้อ 5	ข้อ 6	ข้อ 7	ข้อ 8	ข้อ 9	ข้อ 10
ข้อ 11	ข้อ 12	ข้อ 13	ข้อ 14	ข้อ 15	ข้อ 16	ข้อ 17	ข้อ 18	ข้อ 19	ข้อ 20

- 5.1. ข้อใดคือการอธิบายที่ดีที่สุดสำหรับความซับซ้อนของเวลาในรูปแบบ Big O notation?

ก. ขอบเขตล่างของเวลาที่ใช้ในการทำงานของอัลกอริทึม
 ข. ขอบเขตบนของเวลาที่ใช้ในการทำงานของอัลกอริทึม
 ค. เวลาทำงานเฉลี่ยของอัลกอริทึม
 ง. เวลาทำงานที่แน่นอนของอัลกอริทึม

- 5.2. กำหนดให้ฟังก์ชัน $f(n) = n^2$ และ $g(n) = n^2 \log n$, ข้อใดเป็นความจริงเกี่ยวกับพฤติกรรมอัตราการเติบโตเมื่อ n มีค่ามากขึ้น?

ก. $f(n)$ มีอัตราการเติบโตมากกว่า $g(n)$
 ข. $g(n)$ มีอัตราการเติบโตมากกว่า $f(n)$
 ค. $f(n)$ และ $g(n)$ ไม่สามารถเปรียบเทียบอัตราการเติบโตได้
 ง. $f(n)$ และ $g(n)$ มีอัตราการเติบโตเท่ากัน

- 5.3. ถ้าอัลกอริทึมมีความซับซ้อนของพื้นที่เป็น $O(n)$ และ ความซับซ้อนของเวลาเป็น $O(\log n)$, ข้อใดเป็นความจริง?

ก. อัลกอริทึมต้องการเวลามากขึ้นเมื่อขนาดของข้อมูลนำเข้าเพิ่มขึ้น แต่พื้นที่ใช้งานคงที่
 ข. อัลกอริทึมต้องการพื้นที่มากขึ้นเมื่อขนาดของข้อมูลนำเข้าเพิ่มขึ้น แต่เวลาเพิ่มขึ้นเป็นแบบลอการิทึม
 ค. ทั้งพื้นที่ที่ต้องการและเวลาเพิ่มขึ้นเป็นเส้นตรงตามขนาดของข้อมูลนำเข้า
 ง. อัลกอริทึมใช้พื้นที่ไม่คุ้มค่าเมื่อเทียบกับประสิทธิภาพด้านเวลา

- 5.4. หาก m เป็น `CP::map_bst` ที่มีข้อมูล n ตัว Code ต่อไปนี้ มีความซับซ้อนเชิงเวลาเป็นอะไร?

```
int sum = 0;
while (n--) {
    for (auto it = m.begin(); it != m.end(); it++)
        sum++;
}
```

ก. $O(n)$
 ข. $O(n \log n)$
 ค. $O(n^2)$
 ง. $O(n^2 * \log n)$

- 5.5. เมื่อใช้ priority queue กับ comparator, comparator กำหนดอะไร?

ก. ขนาดของ priority queue
 ข. โครงสร้างข้อมูลพื้นฐานของ priority queue
 ค. ลำดับที่ข้อมูลจะถูกลบออกจาก queue
 ง. จำนวนข้อมูลที่สามารถแทรกเข้าไปใน queue

- 5.6. พิจารณา priority queue ที่ได้รับการดำเนินการด้วย binary heap หลังจากที่เราลบข้อมูลที่มีความสำคัญสูงสุดออกไปแล้ว จะต้องทำอย่างไรเพื่อรักษาคุณสมบัติของ heap?

ก. ต้องพิจารณาข้อมูลทั้งหมดใน binary heap ว่าเป็นไปตามคุณสมบัติของ heap หรือไม่
 ข. ข้อมูลสุดท้ายของ heap จะถูกย้ายไปที่รากและจากนั้นปรับ heap จากด้านบนลงล่าง
 ค. สลับข้อมูลที่รากกับข้อมูลสูงสุดใน heap
 ง. ต้องแทรกข้อมูลใหม่เพื่อแทนที่ข้อมูลที่โดนลบไป

- 5.7. ความซับซ้อนของเวลาในกรณีที่แย่ที่สุดของการค้นหาค่าต่ำสุดใน BST คืออะไร?

ก. $O(1)$
 ข. $O(\log n)$
 ค. $O(n)$
 ง. $O(n \log n)$

- 5.8. คุณสมบัตินี้ทำให้ AVL Tree แตกต่างจาก BST ทั่วไป?

ก. AVL Trees มีความสูงมากกว่า
 ข. AVL Trees มีการรักษาสมดุลของความสูงของต้นไม้
 ค. AVL Trees ไม่อนุญาตให้มีค่าซ้ำกัน
 ง. AVL Trees ใช้การเดินผ่านต้นไม้แบบแตกต่างจาก BSTs ทั่วไป

- 5.9. ใน BST, การลบโหนดที่มีลูกเดียวต่างจากการลบโหนดที่มีลูกสองตัวอย่างไร?

ก. โหนดที่มีลูกเดียวสามารถถูกลบได้โดยตรงโดยไม่ต้องปรับแต่งเพิ่มเติม
 ข. โหนดที่มีลูกเดียวต้องการให้ต้นไม้ย่อยของมันถูกปรับสมดุล
 ค. โหนดที่มีลูกสองตัวถูกแทนที่ด้วยค่าบางค่า
 ง. ไม่มีความแตกต่างในกระบวนการลบสำหรับทั้งสองกรณี

5.10. ใน BST วิธีการเดินผ่านต้นไม้ใดที่คุณจะใช้เพื่อพิมพ์ข้อมูลทั้งหมดในลำดับที่เรียงจากน้อยไปมาก?

- ☒ ก. การเดินผ่านแบบ Inorder
- ข. การเดินผ่านแบบ Preorder
- ค. การเดินผ่านแบบ Postorder
- ง. การเดินผ่านแบบ Level order

5.11. การดำเนินการใดที่อาจต้องการการเดินทางทั่วทั้งต้นไม้ AVL?

- ก. การแทรก
- ☒ ข. การลบ
- ค. การหาความลึกสูงสุด
- ง. การหมุนขวา

5.12. การลบรากของต้นไม้ AVL ที่ไม่ว่างเปล่ามีผลต่อความสูงของต้นไม้อย่างไร?

- ก. ความสูงลดลงหนึ่งขั้นเสมอ
- ข. ความสูงคงที่เสมอ
- ค. ความสูงอาจลดลง คงที่ หรือเพิ่มขึ้น
- ☒ ง. ความสูงอาจลดลงหรือคงที่ แต่ไม่เพิ่มขึ้น

5.13. ในต้นไม้ AVL, สถานการณ์ใดที่จำเป็นต้องมีการหมุนคู่ (ซ้าย-ขวา หรือ ขวา-ซ้าย) ระหว่างการปรับสมดุล?

- ก. เมื่อปัจจัยสมดุลของโหนดกลายเป็น 0
- ☒ ข. เมื่อโหนดลูกมีความไม่สมดุลในทิศทางตรงข้ามกับโหนดพ่อ/แม่
- ค. เมื่อทั้งสองโหนดลูกของโหนดมีความไม่สมดุลเท่ากัน
- ง. เมื่อต้นไม้กลายเป็นไม่สมดุลอย่างสิ้นเชิง

5.14. ผลกระทบของฟังก์ชันแฮชที่ดีต่อประสิทธิภาพของตารางแฮชคืออะไร?

- ก. รับประกันว่าตารางจะไม่เต็มเลย
- ☒ ข. ลดโอกาสของการชนกัน
- ค. เรียงลำดับ key ตามลำดับตัวอักษร
- ง. เข้ารหัส key เพื่อเพิ่มความปลอดภัย

5.15. load factor ของตารางแฮชมีผลต่อประสิทธิภาพอย่างไร?

- ก. load factor ที่สูงขึ้นเร่งเวลาการเข้าถึง
- ☒ ข. load factor ที่สูงขึ้นเพิ่มโอกาสของการชนกัน
- ค. load factor ที่สูงขึ้นต้องการ memory มากขึ้น
- ง. load factor ที่สูงขึ้นลบข้อมูลได้เร็วขึ้น

5.16. ความคิดหลักเบื้องหลังการใช้ separate chaining ในตารางแฮชคืออะไร?

- ก. แต่ละช่องของตารางแฮชมีตัวชี้ไปยังตารางแฮชอื่น
- ☒ ข. แต่ละช่องของตารางแฮชจัดเก็บรายการข้อมูลเพื่อจัดการกับการชนกัน
- ค. ตารางแฮชถูกแบ่งออกเป็นตารางย่อยเล็กๆ เพื่อกระจายข้อมูล
- ง. แต่ละ key ถูกเชื่อมโยงไปยัง key ถัดไปในตารางแฮช

5.17. ในตารางแฮชที่ใช้ separate chaining, ความซับซ้อนของเวลาในกรณีที่แย่ที่สุดสำหรับการค้นหาข้อมูลคืออะไร?

- ก. $O(1)$
- ข. $O(\log n)$
- ☒ ค. $O(n)$
- ง. $O(n \log n)$

5.18. ข้อเสียหลักของการใช้ open addressing เมื่อ load factor ของตารางแฮชสูงคืออะไร?

- ก. การใช้หน่วยความจำเพิ่มขึ้น
- ข. ความยากลำบากในการปรับขนาดตาราง
- ☒ ค. โอกาสที่จะเกิดลำดับการ probe ที่ยาวขึ้น
- ง. ความปลอดภัยที่ลดลงต่อการโจมตีทางไซเบอร์

5.19. การใช้ open addressing จัดการการลบข้อมูลจากตารางแฮชอย่างไร?

- ก. ทำเครื่องหมายช่องที่ลบออกให้เป็นช่องว่าง
- ☒ ข. ทำเครื่องหมายช่องที่ลบออกเป็นช่องที่ถูกลบหรือใช้เครื่องหมายพิเศษ
- ค. ปรับขนาดตารางโดยอัตโนมัติ
- ง. ทั้งช่องนั้นไว้โดยไม่มีการทำเครื่องหมายใดๆ

5.20. ทำไม double hashing ถึงถือว่าเป็นเทคนิคการสำรวจที่มีประสิทธิภาพใน open addressing?

- ☒ ก. มันลดโอกาสของการเกิด secondary clustering
- ข. มันทำให้กระบวนการลบข้อมูลเร็วขึ้น
- ค. มันทำให้การคำนวณฟังก์ชันแฮชเรียบง่ายขึ้น
- ง. มันรับประกันการหาช่องว่างในการสำรวจครั้งเดียว

Part II ให้ทำข้อสอบในระบบ grader

เฉพาะข้อสอบ part II เท่านั้น ต้องทำใน grader เท่านั้น ข้อสอบอยู่ในระบบ Grader แล้ว

Part III ให้ทำข้อสอบในระบบ grader หรือ เขียนคำตอบในหน้าถัดไป

เฉพาะข้อสอบ part III เท่านั้น สามารถเลือกได้ว่าทำใน grader หรือจะทำในกระดาษ หากเลือกทำใน grader คะแนนที่ได้จะเป็นตามระบบ grader เลย หากเลือกทำในกระดาษ คะแนนจะคิดจากคำตอบในกระดาษเท่านั้น เกณฑ์ในการตรวจในกระดาษอาจจะไม่เหมือนเกณฑ์ใน grader ก็ได้ แต่ไม่ว่าจะทำในกระดาษ หรือ grader ก็สามารถได้คะแนนเต็มเท่ากัน ข้อสอบอยู่ในระบบ Grader แล้ว

เลขประจำตัว

--	--	--	--	--	--	--	--	--

ห้องสอบ.....เลขที่ใบใบเซ็นชื่อเข้าสอบ

--	--	--

หน้าที่ 6

หากต้องการทำข้อสอบ Part III โดยใช้วิธีเขียนคำตอบลงกระดาษ แทนการใช้คะแนนจากระบบ Grader ให้ระบุข้อที่ต้องการให้ตรวจในกระดาษลงในหน้านี้ และเขียนคำตอบลงในหน้านี้เป็นต้นไป (หากที่ไม่พอ ให้ใช้เนื้อที่ด้านหลังของหน้านี้เป็นต้นไปได้ด้วย)

ข้าพเจ้าต้องการให้ตรวจข้อสอบ Part III ข้อต่อไปนี้ จากคำตอบที่เขียนในหน้านี้เป็นต้นไป โดยไม่ใช้คะแนนจากระบบ Grader