

Heap Find

จงเขียนฟังก์ชันเพิ่มเติมความสามารถในการหาข้อมูลของ `CP::priority_queue` โดยให้เพิ่มฟังก์ชันสองฟังก์ชันให้กับคลาสดังกล่าว คือ

- `bool find(T k)` ซึ่งจะหาว่าใน `priority_queue` ของเรานั้นมีข้อมูลที่มีค่าเท่ากับ `k` อยู่หรือไม่ โดยจะต้องคืนค่า `true` เมื่อใน `priority_queue` ของเรามีข้อมูลที่มีค่า `k` อยู่เท่านั้น ฟังก์ชันนี้ควรจะใช้เวลาในการทำงานไม่แย่ไปกว่า $O(n)$
- `int find_level(T k)` ฟังก์ชันนี้จะทำงานคล้ายกับ `find` คือจะหาว่าหาว่าใน `priority_queue` ของเรานั้นมีข้อมูลที่มีค่าเท่ากับ `k` อยู่หรือไม่ โดยฟังก์ชันนี้จะคืนค่า "ความลึก" ของปมที่มีข้อมูล `k` อยู่ โดยค่าความลึกคือจำนวนเส้นเชื่อมของ path จากปมรากถึงปมที่มีข้อมูล `k` อยู่ ตัวอย่างเช่น ปมราก นั้นถือว่ามีความลึกเป็น 0, ปมลูกของปมราก จะมีความลึกเป็น 1 และ ปมลูกของปมลูกของรากนั้นจะมีความลึกเป็น 2 เป็นต้น

ในกรณีที่ `priority_queue` ของเรานั้นมีปมหลายปมที่มีค่า `k` ให้ตอบความลึกของปมที่อยู่ "ห่าง" ปมรากมากที่สุด ตัวอย่างเช่น `priority_queue` ที่เกิดจากการใส่ข้อมูล 100, 20, 20, 20 เข้าไปนั้นการเรียก

`find_level(20)` จะต้องคืนค่า 2 ไม่ใช่คืนค่า 1

ในกรณีที่ไม่มีเจอข้อมูลให้คืนค่า -1

ฟังก์ชันนี้ควรจะใช้เวลาในการทำงานไม่แย่ไปกว่า $O(n)$

ข้อบังคับ

ฟังก์ชันทั้งสองจะต้องไม่ทำการแก้ไขข้อมูลใด ๆ ใน `priority_queue`

โจทย์ข้อนี้จะมีไฟล์โปรเจกต์ของ `code::block` ให้ ซึ่งในโปรเจกต์ดังกล่าวจะมีไฟล์ `priority_queue.h`, `main.cpp` และ `student.h` อยู่ ให้นักศึกษาเขียน code เพิ่มเติมลงไปไฟล์ `student.h` เท่านั้น และการส่งไฟล์ขึ้น grader ให้ส่งเฉพาะไฟล์ `student.h`

คำอธิบายฟังก์ชัน main()

`main` จะทำการเรียกใช้ฟังก์ชันต่าง ๆ ของ `stack` ของเรา โดยโปรแกรมจะเริ่มจาก `priority_queue<int>` มา 1 อัน และเรียกใช้งาน `stack` ดังกล่าวตามข้อมูลคำสั่งที่ได้รับจาก keyboard มาทีละบรรทัด แต่ละบรรทัดนั้นจะมีการทำงานต่าง ๆ ซึ่งขึ้นอยู่กับตัวอักษรตัวแรกในบรรทัด โดยที่ `a` เป็นการ push ข้อมูลเข้าไปใน `priority_queue`, `d` เป็นการ pop ข้อมูล, `f` เป็นการเรียกใช้บริการ `find`, `l` เป็นการเรียกใช้ `find_level` และ `q` เป็นการจบการทำงาน

- บรรทัดที่มีคำสั่ง `a` และ `f` และ `r` จะตามด้วยตัวเลข 1 ตัว ซึ่งเป็น parameter ของฟังก์ชันดังกล่าว
- คำสั่ง `f` และ `r` จะพิมพ์ค่าที่ได้รับคืนมาจากการเรียกฟังก์ชันด้วย

ตัวอย่าง

ข้อมูลที่พิมพ์เข้าทาง keyboard	ข้อมูลที่เป็นผลจากการทำงานของโปรแกรม
a 1 a 2 a 3 a 4 a 5 f 5 l 5 d l 5 q	Found 5 Found 5 at level 0 5 not found
a 10 a 10 a 10 a 10 a 20 f 10 l 10 f 20 l 9000 f 9000 d l 10 q	Found 10 Found 10 at level 2 Found 20 9000 not found 9000 not found Found 10 at level 2

คำแนะนำ

- ถ้าทำได้เพียงฟังก์ชันเดียวจากทั้งสองฟังก์ชัน ก็สามารถส่งได้