

List Extract

จงเพิ่มบริการให้กับคลาส `CP::list` โดยให้เพิ่มฟังก์ชัน `void extract(const T& value, iterator a, iterator b, CP::list<T> &output)` ซึ่งจะทาลบข้อมูลทั้งหมดใน list ที่มีค่าเท่ากับ value โดยจะทำการพิจารณาเฉพาะข้อมูลใน list ตั้งแต่ตัวที่ a ซึ่อยู่ไปจนถึงตัวที่อยู่ก่อน b และฟังก์ชันดังกล่าวจะต้องแก้ไข output โดยย้ายข้อมูลทั้งหมดที่ลบออกจาก list ไปใส่ไว้ด้านหน้าของ output

ตัวอย่างเช่น หาก list มีข้อมูลเป็น `<10, -1, 30, -1, 40, -1>` และให้ a ชี้ที่ข้อมูล 10 และ b ชี้ที่ข้อมูล -1 ตัวสุดท้าย และให้ output มีข้อมูลเป็น `<99>` การเรียก `extract(-1,a,b,output)` จะทำให้ข้อมูลใน list กลายเป็น `<10, 30, 40, -1>` และทำให้ output กลายเป็น `<-1,-1,99>`

รับประกันว่าการเรียกฟังก์ชันนี้ a และ b จะชี้ไปยังสมาชิกที่อยู่ใน list ตั้งแต่ `begin()` รวมถึง `end()` และ b จะไม่อยู่ก่อน a แน่แน่นอน และหากเรียกฟังก์ชันนี้ขณะที่ list ไม่มีข้อมูลอยู่ หรือเมื่อ `a == b` ให้ฟังก์ชันนี้ไม่ต้องทำงานใด ๆ

ข้อบังคับ

โจทย์ข้อนี้จะมีไฟล์โปรเจ็คของ `Code::Blocks` ให้ ซึ่งในโปรเจ็คดังกล่าวจะมีไฟล์ `list.h`, `main.cpp` และ `student.h` อยู่ให้นักเขียน code เพิ่มเติมลงไปในไฟล์ `student.h` เท่านั้น และการส่งไฟล์ขึ้น grader ให้ส่งเฉพาะไฟล์ `student.h`

- 50% ของ test data นั้น อนุญาตให้สามารถแก้ไข `student.h` ได้โดยอิสระ สามารถ include และเรียกใช้ data structure หรือ ฟังก์ชัน อื่นใดของ list ได้
- อีก 50% ที่เหลือ ไม่อนุญาตให้ใช้ฟังก์ชัน `insert`, `erase` รวมถึงฟังก์ชันอื่นใดที่เรียกใช้ `insert`, `erase` ของ list

*** ห้ามทำการพิมพ์ข้อมูลทางจอภาพหรืออ่านข้อมูลจากคีย์บอร์ดในไฟล์ `student.h` ที่ส่งมายัง grader โดยเด็ดขาด ***

คำอธิบายฟังก์ชัน main

โปรแกรมจะเริ่มต้นจาก `CP::list<int> l` ซึ่งเป็น list ว่าง และจะรับข้อมูล 2 บรรทัดในรูปแบบต่อไปนี้

- บรรทัดแรกประกอบด้วยจำนวนเต็ม 4 ตัวคือ n, value, a, b
- บรรทัดที่สองประกอบด้วยจำนวนเต็ม n ตัว ซึ่งจะเป็นข้อมูลที่ใส่เข้าไปใน list l ตามลำดับ

หลังจากนั้น main จะสร้าง iterator สองตัวคือ ai และ bi ซึ่งจะชี้ไปยังข้อมูลตำแหน่งที่ a, b ใน list และสร้าง output ซึ่งมีข้อมูลภายในเป็น `<99>` แล้วเรียกใช้ `l.extract(value, ai, bi, output)` และทำการพิมพ์ข้อมูลใน list และ output ออกมาทางหน้าจอ

*** main ใน grader นั้นจะแตกต่างจาก main ที่นิสิตได้รับ แต่จะเป็นการทดสอบในลักษณะเดียวกัน ขอให้เขียนฟังก์ชันเพิ่มเติมให้ตรงตามนิยามที่กำหนดไว้ข้างต้น ***

ตัวอย่าง

ข้อมูลนำเข้า	ข้อมูลส่งออก
6 -1 0 5 10 -1 30 -1 40 -1	Size = 4 From FRONT to BACK: 10 30 40 -1 From BACK to FRONT: -1 40 30 10 Size = 3 From FRONT to BACK: -1 -1 99 From BACK to FRONT: 99 -1 -1