

Change Data

(1 sec, 512mb)

จงเขียนฟังก์ชัน 3 ฟังก์ชันต่อไปนี้ ซึ่งแต่ละฟังก์ชันจะรับข้อมูล 3 ตัวได้แก่ a, from และ to โดยทั้ง 3 ฟังก์ชันจะต้องทำการเปลี่ยนแปลงค่าภายในตัวแปร a ประเภท int ที่มีค่าเป็น from ให้กลายเป็น to ทั้งหมด โดยที่ข้อมูลอื่น ๆ ยังคงต้องมีอยู่ตามเดิม

เราสามารถแก้ไข a และข้อมูลภายในอย่างไรก็ได้ (รวมถึงการลบข้อมูล หรือ เพิ่มข้อมูลเข้าไป) แต่เมื่อกระทำการดังกล่าวเสร็จหมดแล้ว ข้อมูลภายใน a ต้องเป็นไปตามที่กำหนด (คือมีเฉพาะค่า int ที่เป็น from เท่านั้นที่ถูกเปลี่ยนเป็น to การตรวจสอบว่าค่าของ a ถูกเปลี่ยนอย่างถูกต้องหรือไม่จะกระทำโดยการพิมพ์ข้อมูลใน a ออกมาตามลำดับของประเภทข้อมูลนั้น ๆ

ฟังก์ชันทั้ง 3 คือ

- void change_1(stack<vector<queue<int>>> &a, int from, int to)
- void change_2(map<string,pair<priority_queue<int>,int>> &a, int from, int to)
- void change_3(set<pair<list<int>,map<int,pair<int,string>>>> &a, int from, int to)

ข้อบังคับ

- โจทย์ข้อนี้จะมีไฟล์ตั้งต้นมาให้ ประกอบด้วยไฟล์ main.cpp และ student.h อยู่ให้เขียน code เพิ่มเติมลงในไฟล์ student.h เท่านั้น และการส่งไฟล์เข้าสู่ระบบ grader ให้ส่งเฉพาะไฟล์ student.h เท่านั้น
 - ไฟล์ student.h จะต้องไม่ทำการอ่านเขียนข้อมูลใด ๆ ไปยังหน้าจอหรือคีย์บอร์ดหรือไฟล์ใด ๆ
- หากใช้ VS Code ให้ทำการ compile ที่ไฟล์ main.cpp

**** main ที่ใช้จริงใน grader นั้นจะแตกต่างจาก main ที่ได้รับในไฟล์ตั้งต้นแต่จะทำการทดสอบในลักษณะเดียวกัน ****

คำอธิบายฟังก์ชัน main

- บรรทัดแรกประกอบด้วยจำนวนเต็ม 3 ตัวซึ่งระบุจำนวนเต็ม x โดยที่ x มีค่าเป็น 1 2 หรือ 3 เพื่อจะบอกว่าเราต้องการเรียกฟังก์ชัน change_x และค่า from และ to
- หลังจากนั้น main จะเรียกฟังก์ชัน input_x เพื่อรับข้อมูลเข้าสู่ตัวแปร a ตามประเภทดังกล่าว แล้วเรียกฟังก์ชัน change_x(a, from, to)
- หลังจากนั้น main จะเรียกฟังก์ชัน print_x ซึ่งทำการพิมพ์ค่า a ออกมาตามลำดับ

ข้อมูลส่งออก

- พิมพ์ข้อมูลใน a ออกมาตามลำดับของประเภทข้อมูลนั้นๆ โดยสังเกตได้จากตัวอย่างประกอบด้านล่าง

ชุดข้อมูลทดสอบ

- 20% มีเฉพาะการเรียก change_1 เท่านั้น
- 20% มีเฉพาะการเรียก change_2 เท่านั้น และจำนวนข้อมูลทั้งหมดมีไม่เกิน 1000 int
- 10% มีเฉพาะการเรียก change_2 เท่านั้น
- 20% มีเฉพาะการเรียก change_3 เท่านั้น โดยจำนวนข้อมูลทั้งหมดมีไม่เกิน 1000 int และ key ที่มีอยู่แล้วใน map จะไม่เท่ากับ to เลย
- 30% มีเฉพาะการเรียก change_3 เท่านั้น และ key ที่มีอยู่แล้ว ใน map จะไม่เท่ากับ to เลย

ตัวอย่าง

ข้อมูลนำเข้า	ข้อมูลส่งออก
1 1 2 2 // stack size 3 // vector size 2 1 2 // queue size + queue elements 2 1 4 // queue size + queue elements 2 5 6 // queue size + queue elements 1 // vector size 2 7 1 // queue size + queue elements	7,2, 2,2, 2,4, 5,6,

ข้อมูลนำเข้า	ข้อมูลส่งออก
<pre> 2 1 2 3 // map size a 3 1 2 3 // PQ size + PQ elements 6 b 3 7 8 9 // PQ size + PQ elements 1 c 2 1 -20 // PQ size + PQ elements 99 </pre>	<pre> a:3 2 2 ,6 b:9 8 7 ,2 c:2 -20 ,99 </pre>
<pre> 3 1 2 2 // set size 5 1 2 3 4 5 // List size + List elements 3 // map size 1 2 a 3 4 b 5 1 c 4 -9 1 -7 1 // List size + List elements 3 // map size 9 2 dd 8 3 aa 11 4 bb </pre>	<pre> -9 2 -7 2 8:3,aa 9:2,dd 11:4,bb 2 2 3 4 5 2:2,a 3:4,b 5:2,c </pre>