

นับจำนวนข้อมูลใน Queue แบบหลายจำนวน

จงเขียนฟังก์ชันเพิ่มเติมความสามารถให้กับ `CP::queue` โดยให้นิยามฟังก์ชัน `vector<pair<T,size_t>> count_multi(vector<T> k)` ซึ่งจะทำการนับว่าใน queue นั้น มีข้อมูลกี่ตัวที่เท่ากับข้อมูลต่าง ๆ ที่ระบุไว้ใน K กล่าวอีกนัยหนึ่งคือ เราอยากทราบว่า สำหรับแต่ละค่าในตัวแปร k นั้น ค่าดังกล่าวปรากฏใน q เป็นจำนวนกี่ครั้ง โดยฟังก์ชันนี้จะต้องคืนค่าเป็น vector ซึ่งมีขนาดเท่ากับขนาดของ vector k และค่าแต่ละค่าใน vector ที่คืนมานั้นจะประกอบด้วย pair ซึ่งข้อมูลตัวแรกคือค่าของสมาชิกของ k และข้อมูลตัวที่สองคือจำนวนครั้งที่สมาชิกของ k นั้นปรากฏอยู่ใน queue ของเรา โดยลำดับของค่าของสมาชิกของ k ที่คืนมานั้นจะต้องไม่เปลี่ยนแปลง

ตัวอย่างผลการเรียกฟังก์ชัน `q.count_multi` สำหรับค่าของ queue q และ vector k ต่าง ๆ เป็นดังต่อไปนี้

ค่าของ queue q	ค่าของ vector k	ผลลัพธ์ของการเรียก <code>q.count_multi(k)</code>
{1,1,2,2,3}	{1,10,3}	{ {1,2}, {10,0}, {3,1} }
{1,2,3,4,5,1}	{1,1,1}	{ {1,2}, {1,2}, {1,2} }
{1,2,3}	{ }	{ }
{ }	{30,0,-5}	{ {30,0}, {0,0}, {-5,0} }

ข้อบังคับ

โจทย์ข้อนี้จะมีไฟล์โปรเจ็คของ Code::Blocks ให้ ซึ่งในโปรเจ็คดังกล่าวจะมีไฟล์ `queue.h`, `main.cpp` และ `student.h` อยู่ ให้นิยามเขียน code เพิ่มเติมลงในไฟล์ `student.h` เท่านั้น และการส่งไฟล์ขึ้น grader ให้ส่งเฉพาะไฟล์ `student.h` นิยามสามารถแก้ไข `student.h` ได้โดยอิสระ สามารถ include และเรียกใช้ data structure หรือ ฟังก์ชันใด ๆ ได้ *** อย่างไรก็ตาม `count_multi` จะต้องไม่ทำการแก้ไขข้อมูลใน queue โดยเด็ดขาด และห้ามสร้าง queue ใหม่ใด ๆ ขึ้นมาโดยเด็ดขาด ***

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

คำอธิบายฟังก์ชัน `main()`

ฟังก์ชัน `main` จะรับข้อมูลจากคีย์บอร์ดซึ่งจะต้องระบุชุดทดสอบที่ต้องการจะทดสอบ แล้วเรียกฟังก์ชันที่ทดสอบการทำงานของ queue ตามที่กำหนด โดยฟังก์ชันดังกล่าวพิมพ์ผลลัพธ์ของการเรียก `count_multi` ออกมา

*** `main()` ใน grader นั้นจะแตกต่างจาก `main()` ที่นิยามได้รับ ขอให้เขียนฟังก์ชันเพิ่มเติมให้ตรงตามนิยามที่กำหนดไว้ข้างต้น ***