

Stack Move Insert

(ข้อนี้ดัดแปลงมาจากข้อ 8 ของข้อสอบ midterm ปีการศึกษา 2564)

จงเพิ่มบริการ void moveInsert(int k, CP::stack<T>& s, int m) ให้กับ class CP::stack เพื่อทำการย้ายข้อมูล m ตัวบนสุดของกองซ้อน s ไปยังกองซ้อน *this โดยนำไปแทรก ณ ตำแหน่งที่ k นับจากด้านบนของกองซ้อน (k = 0 จะหมายถึงให้วางเหนือข้อมูลทุกตัวของ *this) โดยหาก k มีค่ามากกว่าหรือเท่ากับจำนวนข้อมูลใน *this ก็ให้ไปแทรกได้สุดของกองซ้อน หาก m มากกว่าหรือเท่ากับ s.size() ก็คือให้ย้ายข้อมูลทุกตัว

ในตัวอย่างต่อไปนี้ การเขียนข้อมูล stack จะเขียนในรูปแบบ [a,b,c,...] โดยให้ถือว่าด้านบนสุดของ stack คือตัวขวาสุดในรายการดังกล่าว

ยกตัวอย่างเช่น ให้ s1 เป็น stack ที่มีข้อมูลเป็น [2, 7, 4, 3] และ s2 มีข้อมูลเป็น [10, 6, 9] การเรียก s1.moveInsert(1, s2, 2) จะทำให้ s1 มีข้อมูลเป็น [2, 7, 4, 6, 9, 3] และ s2 มีข้อมูลเป็น [10]

รับประกันว่าค่า k นั้นมีคุณสมบัติ $0 \leq k \leq s1.size()$ และ $0 \leq m$

ข้อบังคับ

- โจทย์ข้อนี้จะมีไฟล์โปรเจ็คของ Code::Blocks ให้ ซึ่งในไฟล์โปรเจ็คดังกล่าวจะมีไฟล์ stack.h, main.cpp และ student.h อยู่ ให้นิสิตเขียน code เพิ่มเติมลงในไฟล์ student.h เท่านั้น และการส่งไฟล์เข้าสู่ระบบ grader ให้ส่งเฉพาะไฟล์ student.h เท่านั้น
 - ในไฟล์ student.h ดังกล่าวจะต้องไม่ทำการอ่านเขียนข้อมูลใด ๆ ไปยังหน้าจอหรือคีย์บอร์ดหรือไฟล์ใด ๆ

คำอธิบายฟังก์ชัน main()

ใน main จะรับค่าจำนวนเต็ม a, b, k และ m จากคีย์บอร์ด หลังจากนั้นจะรับข้อมูลจำนวน a ตัวมาเก็บไว้ใน stack s1 และรับข้อมูลอีก b ตัวมาเก็บไว้ใน stack s2 หลังจากนั้นจะเรียก s1.moveInsertStack(k,s2,m) และพิมพ์ค่าของ s1 และ s2 ออกทางหน้าจอ โดยพิมพ์ค่าจากด้านบนสุดของ stack ลงมา

***** main ที่ใช้จริงใน grader นั้นจะแตกต่างจาก main ที่ได้รับในไฟล์โปรเจ็คเริ่มต้น แต่จะทำการทดสอบในลักษณะเดียวกัน ****

ตัวอย่างการใช้งาน

| s1 และ s2 (top of stack อยู่ด้านขวา) | k | m | s1 และ s2 หลังเรียก s1.moveInsertStack(k,s2,m) |
|---|---|-----|---|
| S1 = [1,2,3,4,5] S2 = [10,20,30] | 0 | 2 | S1 = [1,2,3,4,5,20,30] S2 = [10] |
| S1 = [1,2,3,4,5] S2 = [10,20,30] | 2 | 2 | S1 = [1,2,3,20,30,4,5] S2 = [10] |
| S1 = [1,2,3,4,5] S2 = [10,20,30] | 5 | 2 | S1 = [20,30,1,2,3,4,5] S2 = [10] |
| S1 = [1,2,3,4,5] S2 = [10,20,30] | 1 | 100 | S1 = [1,2,3,4,10,20,30,5] S2 = [] |