List Zigzag

(1 sec, 512mb)

จงเพิ่มบริการ void zigzag(CP::list<T> &ls) ให้กับ CP::list<T> เพื่อรวม node จาก list ที่ส่งเข้ามา (ls) เข้ากับ list ปัจจุบัน โดยสลับกันไปมา โดยฟังก์ชันนี้มีวิธีการทำงานดังต่อไปนี้

- แทรก node จาก ls เข้าไปใน list ปัจจุบันโดยสลับระหว่าง node ใน list ปัจจุบัน และnode ใน ls
- เริ่มต้นด้วย node แรกของ list ปัจจุบัน ตามด้วย node แรกของ ls สลับไปมา
- หาก list ใดหมดก่อน ก็ใส่ node ที่เหลือของอีก list ต่อไปจนหมด
- หลังจากเสร็จสิ้น list ls จะว่างเปล่า เพราะ node ทั้งหมดถูกย้ายไปยัง list ปัจจุบัน

คำอธิบายฟังก์ชัน main

main() จะสร้าง list<int> มาสองตัว คือ l1 และ l2 หลังจากนั้นจะอ่านข้อมูลมา 4 บรรทัด

- บรรทัดแรกประกอบไปด้วยจำนวนเต็ม N ซึ่งระบุจำนวนข้อมูลใน l1
- บรรทัดที่สองประกอบไปด้วยจำนวนเต็ม N จำนวนคือข้อมูลที่จะใส่เข้าไปใน l1 ตามลำดับ
- บรรทัดที่สามประกอบไปด้วยจำนวนเต็ม M ซึ่งระบุจำนวนข้อมูลใน l2
- บรรทัดที่สองประกอบไปด้วยจำนวนเต็ม M จำนวนคือข้อมูลที่จะใส่เข้าไปใน l2ตามลำดับ

หลังจากนั้น main จะเรียก l1.zigzag(l2) และทำการพิมพ์ข้อมูลทั้งหมดใน l1 ออกมา

ชุดข้อมูลทดสอบ

- 10% รับประกันว่า 1 <= N, M <= 6
- 30% รับประกันว่า N == M และข้อมูลที่เก็บใน list เป็นประเภท int
- 10% รับประกันว่า N != M และข้อมูลที่เก็บใน list เป็นประเภท int
- 50% ไม่มีข้อบังคับอื่นใด

ข้อบังคับ

- ในโจทย์ข้อนี้ ห้ามทำการสร้างปมของ list ใหม่ (ซึ่งหมายความว่าจะไม่สามารถใช้วิธีสร้าง list ใหม่ แล้วค่อย ๆ เติมปมเข้าไปได้) แต่อย่างไรก็ตาม 40% ของ testcase จะอนุญาตให้สร้างปมของ list ใหม่ได้
- โจทย์ข้อนี้จะมีไฟล์ตั้งต้นมาให้ซึ่งประกอบด้วยไฟล์ list.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
- หากใช้ VS Code ให้ทำการ compile ที่ไฟล์ main.cpp

**** main ที่ใช้จริงใน grader นั้นจะแตกต่างจาก main ที่ได้รับในไฟล์โปรเจกต์เริ่มต้น แต่จะทำการทดสอบในลักษณะเดียวกัน ****

ตัวอย่าง

ข้อมูลนำเข้า	ข้อมูลส่งออก
3 1 2 3 3 4 5 6	1 4 2 5 3 6
1 1 7 2 3 4 5 6 7 8	1 2 3 4 5 6 7 8
8 1 2 3 4 5 6 7 8 5 65 66 67 68 69	1 65 2 66 3 67 4 68 5 69 6 7 8