

## Stack Compare Reserve

คลาส `CP::stack` นั้นมีแนวทางการเก็บข้อมูลภายในเหมือน `CP::vector` คือมีการจองพื้นที่ของ Dynamic Array เพื่อเก็บข้อมูล โดยการจองพื้นที่อาจจะมากกว่าจำนวนข้อมูลที่มีอยู่ใน stack ก็ได้ กำหนดให้ “พื้นที่สำรอง” คือจำนวนช่องของพื้นที่ใน Dynamic Array ที่ยังไม่ถูกใช้งานในการเก็บข้อมูลของ stack (กล่าวอีกนัยหนึ่ง พื้นที่สำรอง คือ จำนวนครั้งที่เราสามารถเรียกคำสั่ง `push` ได้ โดยที่ยังไม่เกิดการสร้าง Dynamic Array ใหม่แน่ ๆ)

เราต้องการเปรียบเทียบว่า `CP::stack` สองตัวคือ `a` และ `b` ตัวใดมีพื้นที่สำรองมากกว่ากัน จึงเขียนฟังก์ชันเพิ่มเติมให้กับ `CP::stack` คือ ฟังก์ชัน `int compare_reserve(const CP::stack<T> &other)` ซึ่งฟังก์ชันนี้จะต้องคืนค่าดังต่อไปนี้

- คืนค่า -1 หาก stack ปัจจุบันมี “พื้นที่สำรอง” น้อยกว่า other
- คืนค่า 0 หาก stack ปัจจุบันมี “พื้นที่สำรอง” เท่ากับ other
- คืนค่า 1 หาก stack ปัจจุบันมี “พื้นที่สำรอง” มากกว่า other

### ข้อบังคับ

- `compare_reserve` จะต้องไม่ทำการแก้ไขข้อมูลใด ๆ ใน `CP::stack` ทั้งคู่
- โจทย์ข้อนี้จะมีไฟล์โปรเจ็คของ `Code::Blocks` ให้ ซึ่งในไฟล์โปรเจ็คดังกล่าวจะมีไฟล์ `pair.h`, `main.cpp` และ `student.h` อยู่ ให้นักเขียน code เพิ่มเติมลงในไฟล์ `student.h` เท่านั้น และการส่งไฟล์เข้าสู่ระบบ grader ให้ส่งเฉพาะไฟล์ `student.h` เท่านั้น
  - ในไฟล์ `student.h` ดังกล่าวจะต้องไม่ทำการอ่านเขียนข้อมูลใด ๆ ไปยังหน้าจอ หรือคีย์บอร์ดหรือไฟล์ใด ๆ

### คำอธิบายฟังก์ชัน `main()`

`main` จะอ่านข้อมูลจาก `cin` เพื่อสร้าง `CP::stack` ขึ้นมาสองตัว คือ `a` และ `b` หลังจากนั้นจะแสดงผลจากการเรียก `a.compare_reserve(b)` และ `b.compare_reserve(a)` ออกทาง `cout` โดยข้อมูลที่รับเข้ามามีรูปแบบดังต่อไปนี้

บรรทัดที่ 1: จำนวนเต็ม `n` และ `m`

บรรทัดที่ 2: จำนวนเต็ม จำนวน `n` ตัว ซึ่งระบุถึงข้อมูลที่จะถูก `push` เข้า stack `a` ตามลำดับ

บรรทัดที่ 3: จำนวนเต็ม จำนวน `m` ตัว ซึ่งระบุถึงข้อมูลที่จะถูก `push` เข้า stack `b` ตามลำดับ

**\*\*\* `main` ที่ใช้จริงใน grader นั้นจะแตกต่างจาก `main` ที่ได้รับในไฟล์โปรเจ็คเริ่มต้น แต่จะทำการทดสอบในลักษณะเดียวกัน \*\***

### ตัวอย่าง

หากเราทดลองใช้ `main()` ที่มีให้กับข้อมูลที่นำเข้า stack ดังที่ระบุด้านล่างนี้ ผลลัพธ์ของ `a.compare_reserve(b)` จะมีค่าเป็นดังตารางนี้

<code>CP::stack&lt;int&gt; a</code>	<code>CP::stack&lt;int&gt; b</code>	ผลของ <code>a.compare_reserve(b)</code>
1	10	0
1 2	3	0
9 9 9 9	1 2 3	-1
1 2 3 4 5	8 8	1