

## List Explode

(1 sec, 512mb)

จงเพิ่มบริการ `CP::vector<CP::list<T>>` `CP::list<T>::explode()` ให้กับ `CP::list<T>` ซึ่งฟังก์ชันนี้จะทำการสร้าง `vector` ของ `CP::list` ที่มีขนาดของ `vector` เท่ากับขนาดของ `list` ที่เรียกและ `CP::list` แต่ละตัวใน `vector` นั้นจะเป็น `list` ที่มีข้อมูลเพียงตัวเดียว โดยที่ข้อมูลใน `list` ที่อยู่ ณ ช่องหมายเลข `i` ใน `vector` ก็คือข้อมูลที่อยู่ในตำแหน่งที่ `i` ของ `list` ที่เรียก

ตัวอย่างเช่น หาก `list` มีข้อมูลเป็น `{30, 40, 10, 20}` แล้ว และให้ `v` คือผลลัพธ์จากการเรียก `explode` ของฟังก์ชันนี้ เราจะได้ว่า

- `v[0]` คือ `list` ที่มีข้อมูลคือ `{30}`
- `v[1]` คือ `list` ที่มีข้อมูลคือ `{40}`
- `v[2]` คือ `list` ที่มีข้อมูลคือ `{10}`
- `v[3]` คือ `list` ที่มีข้อมูลคือ `{20}`

กล่าวอีกนัยหนึ่ง ฟังก์ชัน `explode` คือการระเบิด `list` ของเราออกมาเป็น `list` ย่อย ๆ ที่แต่ละ `list` จะมีข้อมูลเพียงตัวเดียวซึ่งคือข้อมูลที่เคยอยู่ใน `list` ที่เรียกนั่นเอง

ในการทำงานของฟังก์ชันนี้ การสร้าง `list` ใหม่ใน `vector` นั้นจะต้องเป็นการ “ย้าย” `node` จาก `list` ที่เรียกไปยัง `list` ใหม่ โดยไม่ใช้การ “สร้าง” `node` ใหม่ แล้ว “ลบ” `node` เก่า อย่างไรก็ตาม หากคำตอบที่ส่งมานั้นเป็นการ “สร้าง” แล้ว “ลบ” แล้ว จะได้คะแนนเป็น 60% ของคะแนนเต็มเท่านั้น

## คำอธิบายฟังก์ชัน main

`main()` จะสร้าง `CP::list<int>` จากข้อมูลที่ได้รับมา โดย `main` จะรับข้อมูลสองบรรทัด

- บรรทัดแรกรับจำนวนเต็ม `n` ซึ่งจะบ่งจำนวนข้อมูลใน `list`
- บรรทัดที่สองประกอบด้วยจำนวนเต็ม `n` ตัว ซึ่งคือข้อมูลใน `list`

หลังจาก `main` นำข้อมูลที่ได้รับมาใส่ใน `list` แล้ว `main` จะเรียก `explode` แล้วแสดงจำนวน ข้อมูลใน `vector` พร้อมด้วยข้อมูลของ `list` แต่ละตัวใน `vector`

## ชุดข้อมูลทดสอบ

- 5% `list` ที่เรียกมีข้อมูล 0 ตัว และข้อมูลที่เก็บใน `list` เป็น `int`
- 5% `list` ที่เรียกมีข้อมูล 1 ตัว และข้อมูลที่เก็บใน `list` เป็น `int`
- 10% `list` ที่เรียกมีข้อมูล 2 ตัว และข้อมูลที่เก็บใน `list` เป็น `int`
- 30% `list` ที่เรียกมีข้อมูล  $\leq 10$  ตัว และข้อมูลที่เก็บใน `list` เป็น `int`
- 10% `list` ที่เรียกมีข้อมูล  $\leq 3$  ตัว
- 40% ไม่มีข้อบังคับอื่นใด

## ข้อบังคับ

- โจทย์ข้อนี้จะมีไฟล์โปรเจ็คของ Code::Blocks ให้ ซึ่งในไฟล์โปรเจ็คดังกล่าวจะมีไฟล์ list.h, vector.h, main.cpp และ student.h อยู่ ให้นิสิตเขียน code เพิ่มเติมลงในไฟล์ student.h เท่านั้น และการส่งไฟล์เข้าสู่ระบบ grader ให้ส่งเฉพาะไฟล์ student.h เท่านั้น
  - o ในไฟล์ student.h ดังกล่าวจะต้องไม่ทำการอ่านเขียนข้อมูลใด ๆ ไปยังหน้าจอหรือคีย์บอร์ดหรือไฟล์ใด ๆ
- หากใช้ VS Code ให้ทำการ compile ที่ไฟล์ main.cpp

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

## ตัวอย่าง

ข้อมูลนำเข้า	ข้อมูลส่งออก
0	Vector size is 0
1 1	Vector size is 1 v[0] Size is 1 1
2 1 2	Vector size is 2 v[0] Size is 1 1 v[1] Size is 1 2
4 4 3 3 4	Vector size is 4 v[0] Size is 1 4 v[1] Size is 1 3 v[2] Size is 1 3 v[3] Size is 1 4