

## ปัญหา ลำดับอักษรตั้งต้น (InitCharSequence)

นักโบราณคดีค้นพบลำดับตัวอักษรของอาณาจักรโบราณ เมื่อศึกษาอย่างละเอียดนักโบราณคดีก็ทราบว่าสายอักขระมีการจัดเรียงอย่างมีรูปแบบ โดยเกิดจากลำดับตั้งต้นที่มีความยาวของตัวอักษรไม่น้อยกว่า 2 ตัวอักษร และตัวอักษรในลำดับตั้งต้นนี้จะมีตัวอักษรซ้ำกันเลย แล้วตัวอักษรในลำดับตั้งต้นจะถูกสลับตำแหน่งทำให้เกิดลำดับใหม่แล้วนำมาลำดับที่ได้ใหม่นี้มาต่อกับลำดับตั้งต้น มีการนำลำดับตั้งต้นมาสลับตำแหน่งและต่อเช่นนี้หลายครั้งจนเกิดเป็นลำดับตัวอักษรที่นักโบราณคดีค้นพบ (จำนวนครั้งในการสร้างลำดับใหม่จากลำดับตั้งต้นมีอย่างน้อย 2 ครั้ง)

ปัญหาคือสิ่งที่นักโบราณคดีได้มาคือลำดับของอักษรที่ยาวมาก และไม่รู้ว่ารูปแบบตั้งต้นคืออะไร โปรดช่วยนักโบราณคดีกลุ่มนี้แกะรูปแบบลำดับอักษรตั้งต้น โดยความยาวของลำดับตั้งต้นที่พบมีตั้งแต่ 2 ตัวอักษรขึ้นไป ลำดับที่นักโบราณคดีพบนี้จะมีการดำเนินการกับลำดับตั้งต้นอย่างน้อย 2 รอบมาต่อกัน คือ ลำดับตั้งต้น ต่อด้วย ลำดับผลลัพธ์จากการดำเนินการครั้งที่ 1 ต่อด้วย ลำดับผลลัพธ์จากการดำเนินการครั้งที่ 2 ... ต่อด้วย ลำดับผลลัพธ์จากการดำเนินการครั้งที่ N

ข้อสังเกต ตัวอักษรในลำดับที่นักโบราณคดีพบจะต้องมีตัวอักษรทุกตัวในลำดับตั้งต้น

ตัวอย่างที่ 1 ลำดับคือ 01234562345601456012360123451234560

ลองทายลำดับตั้งต้นเป็นดังนี้

ลำดับตั้งต้นที่ทาย	ความเป็นไปได้	เหตุผล
01	ไม่ได้	เพราะในลำดับที่พบมีตัวอักษร 23456 ทำให้ลำดับตั้งต้นนี้ไม่ครอบคลุม
012	ไม่ได้	เพราะในลำดับที่พบมีตัวอักษร 3456 ทำให้ลำดับตั้งต้นนี้ไม่ครอบคลุม
0123	ไม่ได้	เพราะในลำดับที่พบมีตัวอักษร 456 ทำให้ลำดับตั้งต้นนี้ไม่ครอบคลุม
01234	ไม่ได้	เพราะในลำดับที่พบมีตัวอักษร 56 ทำให้ลำดับตั้งต้นนี้ไม่ครอบคลุม
012345	ไม่ได้	เพราะในลำดับที่พบมีตัวอักษร 6 ทำให้ลำดับตั้งต้นนี้ไม่ครอบคลุม
0123456	ได้	เพราะครอบคลุมทุกตัวอักษรในลำดับที่พบแล้ว
01234560	ไม่ได้	เพราะตัวอักษร 0 ซ้ำกัน 2 รอบ

ดังนั้นลำดับตั้งต้นคือ 0123456

ตัวอย่างที่ 2 ลำดับคือ BZEWUYWUYBZEBZEWUYWUYBZE

ลองทาลำดับตั้งต้นเป็นดังนี้

ลำดับตั้งต้นที่ทาย	ความเป็นไปได้	เหตุผล
BZ	ไม่ได้	เพราะในลำดับที่พบมีตัวอักษร EWUY ทำให้ลำดับตั้งต้นนี้ไม่ครอบคลุม
BZE	ไม่ได้	เพราะในลำดับที่พบมีตัวอักษร WUY ทำให้ลำดับตั้งต้นนี้ไม่ครอบคลุม
BZEW	ไม่ได้	เพราะในลำดับที่พบมีตัวอักษร UY ทำให้ลำดับตั้งต้นนี้ไม่ครอบคลุม
BZEWU	ไม่ได้	เพราะในลำดับที่พบมีตัวอักษร Y ทำให้ลำดับตั้งต้นนี้ไม่ครอบคลุม
BZEWUY	ได้	เพราะครอบคลุมทุกตัวอักษรในลำดับที่พบแล้ว
BZEWUYW	ไม่ได้	เพราะตัวอักษร W ซ้ำกัน 2 รอบ

ดังนั้นลำดับตั้งต้นคือ BZEWUY

รูปแบบข้อมูลเข้า

บรรทัดแรก	ตัวอักษรภาษาอังกฤษตัวพิมพ์ใหญ่หรือตัวเลขเรียงต่อกันโดยไม่มีช่องว่าง มีความยาวไม่เกิน 1000 ตัวอักษร และมีลำดับใหม่ที่เกิดจากลำดับตั้งต้นไม่น้อยกว่า 2 ลำดับและไม่เกิน 20 ลำดับ
-----------	---

รูปแบบผลลัพธ์

ข้อความที่เป็นลำดับตั้งต้นที่ปรากฏในลำดับที่นักโบราณคดีค้นพบ

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
ABCDBCDACDABDABCABCD	ABCD
ZXCDSASXZADCCDXSZAASDCXZ	ZXCDSA
456321789045678903216543789021	4563217890

## เมธอดที่น่าสนใจในคลาส String

Method signature	คำอธิบาย
char charAt(int)	คือค่าตัวอักษร ณ ตำแหน่งที่ i เช่น "Silpakorn".charAt(4) จะได้ค่า 'a'
int compareTo(String)	เปรียบเทียบสตริงเจ้าของเมธอด กับ สตริงในวงเล็บ ให้สตริงเจ้าของเมธอดเป็น text1 และสตริงในวงเล็บเป็น text2 เช่น "ANT".compareTo("CAT") จะได้ -2 เพราะตัว A ลบ ตัว C เมธอดนี้จะเปรียบเทียบรหัสของตัวอักษรตามตำแหน่งตัวต่อตัว
int compareToIgnoreCase(String)	เปรียบเทียบสตริงเจ้าของเมธอด กับ สตริงในวงเล็บ ให้สตริงเจ้าของเมธอดเป็น text1 และสตริงในวงเล็บเป็น text2 เช่น "ant".compareTo("CAT") จะได้ -2 เพราะตัว a ลบ ตัว c เมธอดนี้จะเปรียบเทียบรหัสของตัวอักษรตามตำแหน่งตัวต่อตัวโดยไม่สนใจ ตัวเล็กตัวใหญ่
int indexOf(String)	คืนค่าตำแหน่งของ String ในวงเล็บที่ปรากฏในสตริงเจ้าของเมธอด ถ้ามี หลายตำแหน่งให้คืนตำแหน่งแรกที่ปรากฏ เช่น "MISSISSIPPI".indexOf("SI") จะได้ค่า 3
int indexOf(String,int)	คืนค่าตำแหน่งของ String ในวงเล็บที่ปรากฏในสตริงเจ้าของเมธอด โดย ตำแหน่งเริ่มต้นที่ทำให้เริ่มจากค่า int ที่ปรากฏในวงเล็บที่เป็นอาร์กิวเมนต์ ตัวที่สอง ถ้ามีหลายตำแหน่งให้คืนตำแหน่งแรกที่ปรากฏ เช่น "MISSISSIPPI".indexOf("SI", 4) จะได้ค่า 5
String substring(int i)	คืนค่าสตริงที่ตัดตั้งแต่ตำแหน่ง i ถึงจบสตริง เช่น "MISSISSIPPI".substring(3) จะได้ค่า "SISIPPI"
String substring(int i0,int i1)	คืนค่าสตริงที่ตัดตั้งแต่ตำแหน่ง i0 ถึง i1-1 เช่น "MISSISSIPPI".substring(3,7) จะได้ค่า "SISI"