

ชื่อ-นามสกุล _____ รหัสประจำตัว _____

ปัญหา 1 รวมข้อความอักขรที่มาก่อน (MinMerge) [7 หรือ 12 คะแนน]

[ทรัพยากรการคำนวณ: 1 วินาที, 32 MB]

กำหนดข้อความภาษาอังกฤษตัวพิมพ์ใหญ่มาให้สองตัวคือ **str1** และ **str2** เช่น **NARUTO** และ **OROCHIMARU** เราต้องการรวมสตริงทั้งสองเข้าด้วยกัน โดยสำหรับคู่ตัวอักษรที่ตำแหน่งตรงกันในสตริงทั้งสอง เราจะเลือกตัวอักษรที่มาก่อนตามพจนานุกรมให้อยู่ในผลลัพธ์

ในกรณีที่สตริงยาวไม่เท่ากัน จะทำให้มีตัวอักษรบางตัวในสตริงที่ยาวกว่าไม่มีคู่ ซึ่งเราก็จะให้ตัวอักษรไม่มีคู่นั้นเป็นอักษรที่อยู่ในผลลัพธ์ เช่น ในที่นี้สตริง **OROCHIMARU** ยาวกว่า ทำให้ไม่มีคู่อักษรของ **MARU** และ **MARU** ก็จะอยู่ในคำตอบด้วย ทำให้เราได้สตริงคำตอบจากการรวมข้อความเป็น **NAOCHIMARU**

จงเขียนเมธอดที่รวมข้อความในลักษณะดังกล่าว ซึ่งการเขียนแบบใช้เมธอดเรียกซ้ำ (Recursive) จะได้คะแนนมากที่สุดที่ 12 คะแนน ส่วนคนที่เขียนแบบธรรมดา (ซึ่งไม่ต้องมีเมธอดใดเพิ่มเติมนอกจาก **main** ในกรณีนี้) จะได้คะแนนสูงสุดที่ 7 คะแนน

แผน ก สำหรับคนที่ไม่ต้องการเขียนแบบเรียกซ้ำ หน้าที่ของคุณคือ “จงเขียนโปรแกรมที่สร้างข้อความลำดับอักษรเพิ่มในลักษณะดังกล่าว” (คุณสามารถเขียนทุกอย่างลงในเมธอด **main** หรือจะแยกเมธอดออกมาก็ได้)

แผน ข สำหรับคนที่เลือกที่จะเขียนแบบเรียกซ้ำ จะได้สูงสุด 12 คะแนน โดยข้อกำหนดคือ

1. คุณจะต้องเขียนเมธอดแบบเรียกซ้ำที่มีรูปแบบเป็น
String merge(String str1, String str2) และห้ามใช้ loop
2. เมธอดนี้ไม่มีการรับข้อมูลเข้าใด ๆ เพิ่มเติม และไม่มีการพิมพ์ข้อความใดในนั้น
3. เมธอด **merge** ไม่มีการอ้างถึงข้อมูลที่อยู่นอกเหนือจากพารามิเตอร์และข้อมูลที่ประกาศไว้ภายในตัวเมธอดเอง (คือใช้ได้เฉพาะพารามิเตอร์และสามารถประกาศตัวแปรเพิ่มเติมเฉพาะภายในเมธอดเท่านั้น ห้ามสร้างตัวแปรที่เป็นสมาชิกคลาสเพิ่มเติม)
4. เมธอด **merge** ไม่มีการพิมพ์ผลลัพธ์ใด ๆ ออกมา แต่จะต้องคืนสตริงที่เป็นผลลัพธ์ไปให้เมธอด **main** พิมพ์ผลลัพธ์ดังกล่าวออกมา
5. ตัวโปรแกรมของคุณจะต้องมีลักษณะตามที่ให้ไว้ทางด้านล่างนี้ คุณสามารถแก้ไขโค้ดได้เฉพาะภายในเมธอด merge เท่านั้น (รวมถึงอาจแก้ไขชื่อคลาสได้ ถ้าจำเป็นสำหรับการทำงานกับระบบตรวจอัตโนมัติ)

ชื่อ-นามสกุล _____ รหัสประจำตัว _____

```
import java.util.Scanner;

public class MinMerge {

    String merge(String str1, String str2) {

        // You can enter your code here, and only here.

    }

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);
        String str1 = scan.next();
        String str2 = scan.next();
        MinMerge merger = new MinMerge();
        System.out.println(merger.merge(str1, str2));

    }

}
```

คำแนะนำ 1 ในกรณีที่ต้องการสร้างสตริงใหม่อีกอันที่ตัดตัวอักษรแรกออกไป ให้ใช้เมธอด `substring` และส่งพารามิเตอร์เป็นเลข 1 เช่น ถ้า `str = "SILP"` และเราเขียนว่า `str.substring(1)` มันจะคืนผลลัพธ์มาเป็น `"ILP"` ทั้งนี้ ขอให้เข้าใจด้วยว่า `str` ไม่ได้ถูกเปลี่ยน แต่เราได้สตริงมาเพิ่มอีกตัว

คำแนะนำ 2 ในกรณีที่ต้องการสร้างสตริงใหม่เริ่มที่ตัวอักษรอินเด็กซ์ `k` ก็ทำได้แบบเดียวกันคือเขียนว่า `str.substring(k)`

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
NARUTO OROCHIMARU	NAOCHIMARU	OROCHIMARU NARUTO	NAOCHIMARU
PIKACHU DRATINI	DIAACHI	LIVERPOOL MANUTD	LANERDOOL
NARUTO SASUKE	NARUKE	GERALT VESEMIR	GERALIR

เงื่อนไขการให้คะแนน โปรแกรมจะต้องทำงานถูกต้องอย่างน้อย 50% ของชุดทดสอบ จึงจะได้คะแนน

หมายเหตุ ข้อมูลเข้ายาวไม่เกิน 500 ตัวอักษร