

Advanced Word Search Puzzle

หาคำในตารางคำ (Word Search Puzzle)

แต่พิเศษตรงที่ต้องหาคำที่เรียงติดกันไม่ใช่แค่ตามแนวนอน ตั้ง ทแยง

แต่ต้องหาคำที่ติดกันแบบยึกๆ ย่อๆ อีกด้วย (ตัวอักษรติดกันในที่นี้คือ ตัวอักษรที่ $i+1$ อยู่ด้านบน ซ้าย ล่าง หรือขวา ของตัวอักษรตัวที่ i)

ตัวอย่าง

โจทย์: จงหาคำว่า DRAGON

W (0, 0)	C (0, 1)	O (0, 2)	N (0, 3)	S (0, 4)
U (1, 0)	D (1, 1)	G (1, 2)	Y (1, 3)	L (1, 4)
Z (2, 0)	R (2, 1)	A (2, 2)	P (2, 3)	O (2, 4)
R (3, 0)	T (3, 1)	E (3, 2)	K (3, 3)	S (3, 4)
I (4, 0)	X (4, 1)	F (4, 2)	J (4, 3)	E (4, 4)

ให้เขียนโปรแกรมที่รับจำนวนแถว (n) จำนวนคอลัมน์ (m) ของตารางคำ

ตามด้วยตารางคำ (g) และคำที่ต้องการหา ($word$)

โดยจะต้องบอกว่าคำที่หาอยู่ตั้งแต่ช่องไหนถึงช่องไหน โดยแสดงผลเป็น list of tuple coordinates โดยพิกัดให้เป็นตามตารางตัวอย่างด้านบน คือ index เริ่มจาก 0 ทั้ง column และ row และแต่ละ tuple ให้แสดงผลเป็น (row_index, column_index) จากตัวอย่างด้านบนจะต้องแสดงผล [(1, 1), (2, 1), (2, 2), (1, 2), (0, 2), (0, 3)]

ข้อมูลนำเข้า

บรรทัดแรก จำนวนแถว n และจำนวนคอลัมน์ m คั่นด้วยเว้นวรรค

บรรทัดที่สอง ถึงบรรทัดที่ $n+1$ ตารางคำ g ในแต่ละแถว

ซึ่งแต่ละแถวจะมีตัวภาษาอังกฤษพิมพ์ใหญ่ m ตัว อยู่ติดกันทั้งหมด ไม่มีเว้นวรรค

บรรทัดที่ $n+2$ รับ **word** ซึ่งเป็นคำที่ต้องการให้หา

เป็นตัวพิมพ์ใหญ่ทั้งหมดเช่นเดียวกัน

ข้อมูลส่งออก

แสดงผลเป็น List of tuple ซึ่ง List จะมีความยาวเท่ากับความยาวของ word

และในแต่ละ tuple จะเก็บค่าพิกัดของตัวอักษร โดยเก็บแบบ (row_index, column_index)

**** หมายเหตุ** รับประกันว่าตารางคำ g จะมี word ที่หาได้แน่ๆและจะมีเพียงคำตอบเดียวเท่านั้น

input (จากแป้นพิมพ์)	output (ทางจอภาพ)
3 4 DAYP DSTZ NOHJ PYTHON	[(0, 3), (0, 2), (1, 2), (2, 2), (2, 1), (2, 0)]
4 6 UHRVTG LCDKQH AILEUJ RKWSRH CHULA	[(1, 1), (0, 1), (0, 0), (1, 0), (2, 0)]
5 7 HCVXYL NWOXYEC OPROANG SXREIGC JYJENDD ENGINEER	[(1, 5), (2, 5), (3, 5), (3, 4), (4, 4), (4, 3), (3, 3), (3, 2)]