

Giant Checker

รูปข้างขวานี้คือตารางขนาด 3x3 ตำแหน่งของแต่ละช่องถูกกำหนดโดยใช้ตัวอักษร 1 ตัว (a, b และ c) แทนแถว ตามด้วยตัวเลขหนึ่งตัว (1, 2, และ 3) แทนคอลัมน์ และโปรแกรมข้างล่างนี้รับ ตำแหน่งช่อง (เช่น a3) เพื่อแสดงสีพื้นของช่องนั้น

c	c1	c2	c3
b	b1	b2	b3
a	a1	a2	a3
	1	2	3

```
position = input().strip()
row = position[0]
col = position[1]
r = 'abc'.find(row)
if r%2 == int(col)%2 :
    print('Black')
else:
    print('White')
```

หมายเหตุ: โปรแกรมข้างบนนี้ ไม่ได้ตรวจสอบกรณีที่ป้อนตำแหน่งผิดไปจากที่ปรากฏในตาราง

สำหรับโปรแกรมที่ต้องเขียนในข้อนี้ ขอขยายตารางเป็นขนาด 52x52 มี a ถึง z และ A ถึง Z เป็นตำแหน่งแถว และ 1 ถึง 52 เป็นตำแหน่งคอลัมน์ (ดูรูปทางขวานี้) จงเขียนโปรแกรมที่รับ ตำแหน่งช่อง เพื่อแสดงสีพื้นของช่องนั้น โดยต้องตรวจสอบความถูกต้องของตำแหน่งช่องด้วย

ข้อมูลนำเข้า

บรรทัดเดียวเป็นสตริง แทนตำแหน่งช่องในตาราง (มีขนาดอย่างน้อยหนึ่งตัวอักษร ตัวแรกไม่ใช่ blank แน่ๆ) ลักษณะข้อมูลนำเข้ามี 2 รูปแบบ

- แบบที่ 1 (มี 80% ของกรณีทดสอบ): หลัง strip แล้วมีไม่เกิน 3 ตัว ตัวแรกแทนตัวอักษรแถว และตัวที่เหลือแทนเลขคอลัมน์ เช่น **A2**
- แบบที่ 2 (มี 20% ของกรณีทดสอบ): หลัง strip แล้วมี 3 ตัวขึ้นไป จะมีรูปแบบ **row = ตัวอักษรแถว, col = เลขคอลัมน์** โดย
 - ส่วนที่แสดงด้วยพื้นเขียวมีใน input แน่นนอน และมีปรากฏอย่างละหนึ่งครั้งแน่ ๆ (ยกเว้น = มีสองครั้ง) ตามลำดับที่แสดง (ยกเว้นคำว่า row กับ col อาจสลับตำแหน่งกันได้)
 - ช่องว่างคั่นมีได้ตั้งแต่ 0 ตัวขึ้นไป
 - ส่วนตัวอักษรแถวและเลขคอลัมน์ อาจไม่มีหรือผิดได้

ข้างล่างนี้แทนตำแหน่ง **A2** ที่ถูกต้องทั้งสิ้น (สำหรับกรณี input ผิด ๆ ดูได้จากตัวอย่างข้างล่าง)

col = 2 , row = A	col= 2, row =A	col =0002 , row = A
row= A, col = 02	col=2,row=A	col = 002 , row = A

(ข้อแนะนำ: ให้ทำแบบที่ 1 ให้เสร็จก่อน แล้วค่อยทำแบบที่ 2)

ข้อมูลส่งออก

- ถ้าตัวอักษรแถวไม่ถูกต้องหรือไม่มี (แต่เลขคอลัมน์ถูกต้อง) ให้แสดง **Invalid row** (ถูกต้องต้องเป็น a ถึง z, A ถึง Z)
- ถ้าเลขคอลัมน์ไม่ถูกต้องหรือไม่มี (แต่ตัวอักษรแถวถูกต้อง) ให้แสดง **Invalid column** (ถูกต้องต้องเป็น 1 ถึง 52)
- ถ้าทั้งตัวอักษรแถวและเลขคอลัมน์ไม่ถูกต้องหรือไม่มี ให้แสดง **Invalid row and column**
- ถ้าตัวอักษรแถวและเลขคอลัมน์ถูกต้อง ให้แสดง **Black** หรือ **White** ให้ตรงกับสีพื้นของตำแหน่งนั้นในตาราง

input (จากแป้นพิมพ์)	output (ทางจอภาพ)
r3w	Invalid column
AA	Invalid column
z76	Invalid column
9 9	Invalid row
\$09	Invalid row
\$ \$	Invalid row and column
()	Invalid row and column
\$59	Invalid row and column
%0X	Invalid row and column
b2	White
z03	Black

input (จากแป้นพิมพ์)	output (ทางจอภาพ)
row= r , col = 3 1	Invalid column
row = A, col = A	Invalid column
col = 76 , row=z	Invalid column
row = 9 , col = 9	Invalid row
row = AAA ,col=09	Invalid row
row = , col = \$\$	Invalid row and column
row =(, col =)	Invalid row and column
col = 59 , row=\$	Invalid row and column
col= 1 1,row=%	Invalid row and column
col = 2 ,row = b	White
row=z, col = 03	Black

ข้อแนะนำ

นิสิตสามารถนำโครงโปรแกรมข้างล่างนี้ไปใช้เป็นแนวทางหนึ่งในการเขียนโปรแกรม

```
pos = input().strip() # รับมาอย่างน้อย 1 ตัวนะ ๑

row = ''
col = ''
if len(pos) <= 3:
    # แบบที่ 1
    row = ???
    col = ???
else:
    # แบบที่ 2 (กรณีนี้ค่อยทำทีหลังก็ได้ เพราะยุ่งกว่า แต่ให้คะแนนแค่ 20%)
    # หาค่าของตัวอักษรแถว มาใส่ในตัวแปร row ถ้าหาไม่พบ ก็ให้เป็นสตริงว่าง ๑
    # หาค่าของเลขคอลัมน์ มาใส่ในตัวแปร col ถ้าหาไม่พบ ก็ให้เป็นสตริงว่าง ๑

    row = row # ในกรณีที่ยังไม่ได้เขียนแบบที่ 2 ก็เขียนคำสั่งนี้ไว้ก่อน (ไม่งั้น ระบบจะฟ้องว่าหลัง else ไม่มีคำสั่ง)

row = row.strip()
col = col.strip()
#-----
valid_row = True
valid_col = True

# ตรวจสอบค่าของสตริง row ถ้าผิด ก็เปลี่ยนตัวแปร valid_row ให้เป็น False

# ตรวจสอบค่าของสตริง col ถ้าผิด ก็เปลี่ยนตัวแปร valid_col ให้เป็น False

#-----
# ตรวจสอบค่าของตัวแปร valid_row และ valid_col
# เพื่อแสดงข้อผิดพลาด ถ้ามีค่าเป็น False (มี 3 กรณี)
# ถ้าไม่มีข้อผิดพลาด ก็ค่อยตัดสินใจว่าจะแสดง Black หรือ White ตามสีพื้นของช่อง row, col
```



[ดาวน์โหลดโครงโปรแกรม](#)