

การเล่นเกเป่ยิงฉุบแบบทีม เป็นการเล่่นเป่ยิงฉุบแบบเดิม แต่ปรับกติกาให้เป็นแบบทีมโดยมีกติกาดังนี้

- ผู้เล่นแต่ละทีมมีจำนวนเท่ากัน และเรียงตามลำดับที่กำหนดไว้
- ในแต่ละรอบ แต่ละทีมจะส่งผู้เล่นเข้ามาตามลำดับที่เรียงไว้
- ตัวแทนของทีมจะเล่่นเป่ยิงฉุบกับตัวแทนจากอีกทีมหนึ่ง ผู้ชนะ 2 ครั้งก่อนอีกฝ่ายหนึ่งจะชนะ และผู้แพ้จะถูกดึงออกจากทีม
- รอบต่อไปจะเล่่นโดยที่ผู้ชนะจากรอบก่อนหน้า กับผู้เล่นในลำดับถัดไปของอีกทีม และหาผู้ชนะ 2 ครั้งก่อนอีกฝ่ายหนึ่ง
- เกมจะเล่่นกันไปเรื่อย ๆ จนทีมใดทีมหนึ่งไม่เหลือผู้เล่นเลย

ในโจทย์ข้อนี้ ต้องการให้เขียนฟังก์ชัน `player_moves()` ในการสรุปว่าผู้เล่นแต่ละคนออกอะไรไปบ้าง

```
def player_moves(playing_order, team_a, team_b):  
    # TODO: Write your code here!  
  
    return team_a_moves, team_b_moves  
  
# TODO: Do not modify the code below this line  
exec(input().strip())
```

โดยฟังก์ชัน `player_moves()` มีรายละเอียดดังนี้

- ฟังก์ชัน `player_moves()` จะรับ input เป็น `player_order`, `team_a` และ `team_b` และจะคืนค่าเป็น `team_a_moves` และ `team_b_moves` ซึ่งทั้งคู่เป็น list ที่ประกอบไปด้วย list ย่อยที่มีรายละเอียดเป็นชื่อผู้เล่น และท่าที่ผู้เล่นออกทั้งหมด ของแต่ละทีม
- `player_order` คือข้อความที่เก็บข้อมูลการเล่นเกม ลำดับการเก็บจะสลับกันระหว่างผู้เล่นจากแต่ละทีม โดยแต่ละตัวอักษรจะประกอบด้วย R, S และ P หมายถึง ค้อน กรรไกร และกระดากตามลำดับ
- `team_a` คือ list ของข้อความเก็บลำดับชื่อผู้เล่นของทีม A
- `team_b` คือ list ของข้อความเก็บลำดับชื่อผู้เล่นของทีม B

### ตัวอย่างการใช้งานที่ 1

```
player_moves("RSSP", ["Anna", "Alex", "Alice"], ["Ben", "Bella", "Blake"])
```

การเรียกใช้งานคำสั่งนี้ หมายถึงว่า

- Team A ประกอบไปด้วย Anna, Alex และ Alice เรียงลำดับการเล่นตามนี้ เริ่มจาก Anna ก่อน
- Team B ประกอบไปด้วย Ben, Bella และ Blake เรียงลำดับการเล่นตามนี้ เริ่มจาก Ben ก่อน
- ลำดับการเล่นคือ **RSSP**

Team A			Team B		
ผู้เล่น	ท่าที่ออก	คะแนน	คะแนน	ท่าที่ออก	ผู้เล่น
Anna	R (ค้อน)	1	0	S (กรรไกร)	Ben
Anna	S (กรรไกร)	2	0	P (กระดาด)	Ben

จากตารางจะพบว่า

- Team A: ผู้เล่น Anna ออกท่า R (ค้อน) และ S (กรรไกร) ตามลำดับ
- Team B: ผู้เล่น Ben ออกท่า S (กรรไกร) และ P (กระดาด) ตามลำดับ

ดังนั้นการเรียกใช้คำสั่งดังกล่าว จะคืนค่านี้ออกมา

```
([["Anna", ["R", "S"]], [Ben, ["S", "P"]]])
```

## ตัวอย่างการใช้งานที่ 2

```
player_moves("RSSPRRPRPSRS", ["Anna", "Alex", "Alice"], ["Ben", "Bella", "Blake"])
```

การเรียกใช้งานคำสั่งนี้ หมายถึงว่า

- Team A ประกอบไปด้วย Anna, Alex และ Alice เรียงลำดับการเล่นตามนี้ เริ่มจาก Anna ก่อน
- Team B ประกอบไปด้วย Ben, Bella และ Blake เรียงลำดับการเล่นตามนี้ เริ่มจาก Ben ก่อน
- ลำดับการเล่นคือ **RSSPRRPRPSRS**

Team A			Team B		
ผู้เล่น	ท่าที่ออก	คะแนน	คะแนน	ท่าที่ออก	ผู้เล่น
Anna	R (ค้อน)	1	0	S (กรรไกร)	Ben
Anna	S (กรรไกร)	2	0	P (กระดาด)	Ben
Anna ชนะ Ben และทำให้ Ben ต้องออกจากเกม					
Anna	R (ค้อน)	0	0	R (ค้อน)	Bella
Anna	P (กระดาด)	1	0	R (ค้อน)	Bella
Anna	P (กระดาด)	1	1	S (กรรไกร)	Bella
Anna	R (ค้อน)	2	1	S (กรรไกร)	Bella

จากตารางจะพบว่า

- Team A: ผู้เล่น Anna ออกท่า R, S, R, P, P และ R ตามลำดับ
- Team B: ผู้เล่น Ben ออกท่า S และ P ตามลำดับ
- Team B: ผู้เล่น Bella ออกท่า R, R, S และ S ตามลำดับ

ดังนั้นการเรียกใช้คำสั่งดังกล่าว จะคืนค่านี้ออกมา

```
([["Anna", ["R", "S", "R", "P", "P", "R"]],  
[Ben, ["S", "P"]], [Bella, ["R", "R", "S", "S"]]  
)
```

## Input (ข้อมูลนำเข้า)

ข้อมูลนำเข้า มีทั้งหมด 1 บรรทัด มีลักษณะเป็นคำสั่งในภาษา Python ซึ่งเป็นคำสั่งแสดงผลการใช้งานฟังก์ชัน `player_moves()`

## Output (ผลลัพธ์)

แสดงผลที่ได้จากการคืนค่าของฟังก์ชัน `player_moves()` ซึ่งประกอบด้วย `team_a_moves` และ `team_b_moves` โดยทั้งคู่เป็น list ที่ประกอบไปด้วย list ย่อย ที่มีรายละเอียดเป็นชื่อผู้เล่น และท่าที่ผู้เล่นออกทั้งหมด ของแต่ละทีม

## Examples (ตัวอย่าง)

### ตัวอย่างที่ 1

Input
<pre>print(player_moves("RSSP", ["A1", "A2", "A3"], ["B1", "B2", "B3"]))</pre>
Output
<pre>([['A1', ['R', 'S']], [['B1', ['S', 'P']]])</pre>

### ตัวอย่างที่ 2

Input
<pre>print(player_moves("RSSPRR", ["A1", "A2", "A3"], ["B1", "B2", "B3"]))</pre>
Output
<pre>([['A1', ['R', 'S', 'R']], [['B1', ['S', 'P']], ['B2', ['R']]])</pre>

### ตัวอย่างที่ 3

Input
<pre>print(player_moves("RSSPRRPR", ["A1", "A2", "A3"], ["B1", "B2", "B3"]))</pre>
Output
<pre>([['A1', ['R', 'S', 'R', 'P']], [['B1', ['S', 'P']], ['B2', ['R', 'R']]])</pre>

### ตัวอย่างที่ 4

Input
<pre>print(player_moves("RSSPRRPRPSRS", ["A1", "A2", "A3"], ["B1", "B2", "B3"]))</pre>
Output
<pre>([['A1', ['R', 'S', 'R', 'P', 'P', 'R']], [['B1', ['S', 'P']], ['B2', ['R', 'R', 'S', 'S']]])</pre>

### ตัวอย่างที่ 5

Input
<pre>print(player_moves("RSSPRRPRPSRSRPSR", ["A1", "A2", "A3"], ["B1", "B2", "B3"]))</pre>
Output
<pre>([['A1', ['R', 'S', 'R', 'P', 'P', 'R', 'R', 'S']], [['B1', ['S', 'P']], ['B2', ['R', 'R', 'S', 'S']], ['B3', ['P', 'R']]])</pre>