

FRA 141 การเขียนโปรแกรมคอมพิวเตอร์สำหรับวิศวกรรมหุ่นยนต์และระบบอัตโนมัติ 1

(Computer Programming for Robotics and Automation Engineering I)

ภาคเรียนที่ 1 ปีการศึกษา 2565

สถาบันวิทยาการหุ่นยนต์ภาคสนาม มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

หัวข้อ: การแก้ปัญหาด้วยการเขียนโปรแกรมภาษาไพธอน

ส่งงานที่: submit.fibo.kmutt.ac.th

วิธีการสอบ

เขียนฟังก์ชัน ต่อไปนี้ในภาษาไพธอน

* เขียนชื่อฟังก์ชันให้ถูกต้อง ตัวพิมพ์เล็กพิมพ์ใหญ่ต้องให้เหมือนกับโจทย์ ถ้าชื่อฟังก์ชันผิด แล้วเทสไม่ผ่านจะเสียคะแนน

* โจทย์จะกำหนดว่าให้ฟังก์ชัน return ผลลัพธ์อะไรออกมา ใช้คำสั่ง return ตามโจทย์ อย่าใช้วิธี print ผลลัพธ์ออกมา เพราะจะเทสไม่ผ่านและเสียคะแนน

* โจทย์แต่ละข้อจะมีตัวอย่างให้ โดยจะประกอบไปด้วย คำสั่งให้ลองเรียกและผลลัพธ์ที่ควรจะได้ นักศึกษาควรพยายามแก้ไขจนกว่าจะได้ผลลัพธ์ที่ถูกต้องทุกอัน เพื่อให้แน่ใจว่าได้คะแนนเต็ม

* คะแนนเต็ม 10 คะแนน

List# 09

พื้ม่อนต้องการหีบขมจากชั้นวางของแต่หิวมากจึงเดินไปไม่ไหวเลยฝากพี่ออยให้เดินไปหีบแทน ด้วยความที่ชั้นวางของมีความซับซ้อนมาก โดยชั้นวางของนี้ประกอบด้วยช่องย่อย ๆ ซึ่งมีขมวางอยู่ไม่เท่ากัน ซึ่งชั้นวางของอยู่ในรูป list และปริมาณขมคือตัวเลขที่อยู่ใน list เช่น [1] หมายถึงมีชั้นวางของช่องเดียวและมีขมอยู่ 1 ชิ้น, [[1], [3, 2]] หมายถึงชั้นวางของที่มี 2 ช่องใหญ่และช่องด้านขวาแบ่งเป็นช่องย่อยอีก 2 ช่อง

พื้ม่อนต้องการหีบขมทั้งหมดแต่หีบจากมากไปน้อย พื้ม่อนเลยอยากให้น้อง ๆ ช่วยเขียนฟังก์ชัน wantSnack(list) เพื่อเรียงลำดับการหีบขมให้พี่ออย โดยระบุเป็นตำแหน่งของขมบนชั้นวางของให้ได้ปริมาณขมเรียงจากมากไปน้อย เช่น [6, [5, 7]] จะต้องหีบเลข 7 -> 6 -> 5 ซึ่งการหีบเลข 7 ต้องหีบจากตำแหน่ง [1, 1], จะหีบเลข 6 ต้องหีบจากตำแหน่ง 0 (หีบจากชั้นวางของที่ไม่มีช่องย่อยไม่ต้องใส่ [] แต่ถ้าหีบจากชั้นวางของที่มีช่องย่อยต้องใส่ [] ให้เป็นคู่อันดับ) จากนั้น return ออกมาเป็น list ของตำแหน่งเหล่านี้

ปล. ปริมาณขมในแต่ละช่องไม่ซ้ำกัน และอาจได้ตำแหน่งออกมาในรูปตำแหน่ง หรือ [ตำแหน่งชั้นนอก, ตำแหน่งชั้นใน] ขึ้นกับจำนวนช่องของชั้นวางของ

ปล2. ถ้าไม่มีขมเลยให้ return []

เช่น

```
wantSnack([1, 3, 4, 8, 2, 5, 9])
```

```
>>> [6, 3, 5, 2, 1, 4, 0]
```

เนื่องจากต้องหีบขมจาก 9 -> 8 -> 5 -> 4 -> 3 -> 2 -> 1 ซึ่งอยู่บนชั้นตำแหน่งที่ 6, 3, 5, 2, 1, 4, 0 ตามลำดับ

```
wantSnack([1, 2, [4, 5]])
```

```
>>> [[2, 1], [2, 0], 1, 0]
```

เนื่องจากต้องหีบขมจาก 5 -> 4 -> 2 -> 1 ซึ่งอยู่บนชั้นตำแหน่งที่ (2, 1), (2, 0), 1, 0 ตามลำดับ