

Bonus Points

จงเขียนฟังก์ชัน `update(points, uses, reward)`

- `points` เป็นดิก เก็บข้อมูลในรูปแบบ { รหัสสมาชิก: จำนวนแต้มสะสม (ก่อนหักแต้มที่ใช้ไป) }
- `uses` เป็นดิก เก็บข้อมูลในรูปแบบ { รหัสสมาชิก: จำนวนแต้มที่ใช้ไป }
- `reward` เป็นจำนวนเต็ม แทนร้อยละที่ใช้คำนวณแต้มโบนัสที่จะเพิ่มให้สมาชิก
แต้มโบนัสคำนวณจาก แต้มสะสมที่เหลือ (หลังหักแต้มที่ใช้ไป) คูณ `reward` หารด้วย 100 แล้วปัดเศษขึ้น
- ฟังก์ชันนี้ไม่คืนอะไร เพราะผลที่ได้คือการ **ปรับข้อมูลใน `points`** (คือหักแต้มที่ใช้ไป กับเพิ่มแต้มโบนัส)

เช่น มีสมาชิกทั้งหมด 3 คน `points = {'A101': 1000, 'A102': 500, 'A103': 200}`

มีสมาชิก 2 คนใช้แต้มไป `uses = {'A101': 100, 'A103': 50 }`

คำสั่ง `update(points, uses, 10)`

จะทำให้ `points` เปลี่ยนไปเป็น `{ 'A101': 990, 'A102': 550, 'A103': 165 }`

$$(1000 - 100) + 10\% \text{ ของ } (1000 - 100)$$

$$500 + 10\% \text{ ของ } 500$$

$$(200 - 50) + 10\% \text{ ของ } (200 - 50)$$

อย่าลืมว่า การคำนวณแต้มโบนัสที่คิดเป็นเปอร์เซ็นต์นั้น จะปัดเศษหลังจุดทศนิยมทิ้ง (ถ้ามี)

```
def update(points, uses, reward):
```

```
    exec(input().strip()) # ต้องมีบรรทัดนี้เมื่อส่งไป grader
```

ข้อมูลนำเข้า

คำสั่งในการทดสอบฟังก์ชันที่เขียน

ข้อมูลส่งออก

ผลที่ได้จากคำสั่งที่ป้อนเป็นข้อมูลนำเข้า

ตัวอย่าง

input (จากแป้นพิมพ์)

```
p={'A1':100,'A2':50,'A3':20};u={'A1':10,'A3':5};update(p,u,10);print(p)
```

output (ทางจอภาพ)

```
{ 'A1': 99, 'A2': 55, 'A3': 16 }
```