Render

ข้อนี้ต้องถามว่าหากต้องแบ่งหุงข้าว N จาน โดยแต่ละครั้งหุงได้เกิน K จานต่อครั้ง จะสามารถแบ่งหุงจนครบ N ได้กี่วิธี

สำหรับข้อนี้วิธีแรกที่เห็นได้ง่ายที่สุดคือ Dynamic Programming โดยเห็นได้ว่า $dp[i] = \Sigma_{j=\max(0,i-k)}^i dp[j]$ (ในการหุงครั้งจากต้องการ i จะเหลือต้องการ j จานโดย $i-k\leq j\leq i-1$)

แต่เนื่องจากข้อนี้กำหนด $N \leq 100000, K \leq 100000$ วิธีนี้ที่ใช้เวลา $\mathcal{O}(NK)$ จึงช้าเกินไป เราจึงต้องใช้วิธีที่เร็วกว่านี้

เราสามารถกำหนด $C[i] = \Sigma_{j=0}^i dp[j]$ ซึ่งจะทำให้สามารถคำนวณ $dp[i] = C[i-1]$ เมื่อ $i-k-1 < 0$ และ $dp[i] = C[i-1] - C[i-k-1]$ เมื่อ $i-k-1 \geq 0$ ในเวลา $\mathcal{O}(1)$ ($C[i]$ สามารถคำนวณเป็น $C[i]=C[i-1] + dp[i]$ ในเวลา $\mathcal{O}(1)$ เช่นกัน)

ดังนั้นจึงต้องใช้เวลาเพียง $\mathcal{O}(1)$ สำหรับการคำนวณ $C[i]$ และ $dp[i]$ สำหรับแต่ละ $i \leq N$ ทั้งหมดจึงใช้เวลา $\mathcal{O}(N)$

โค้ดประกอบคำอธิบาย

```cpp
dp[0] = 1;
C[0] = 1;
for(int i=1;i<=n;i++)
{
    if (i-k-1>=0)
        dp[i] = C[i-1] - C[i-k-1];
    else
        dp[i] = C[i-1];
    dp[i] = dp[i] %2009;
    if(dp[i]<0)
        dp[i]+=2009;
    C[i] = C[i-1] + dp[i] %2009;
}
```

ข้อนี้ต้องถามว่าหากต้องแบ่งหุงข้าว N จาน โดยแต่ละครั้งหุงได้เกิน K จานต่อครั้ง จะสามารถแบ่งหุงจนครบ N ได้กี่วิธี

สำหรับข้อนี้วิธีแรกที่เห็นได้ง่ายที่สุดคือ Dynamic Programming โดยเห็นได้ว่า $dp[i] = \Sigma_{j=\max(0,i-k)}^i dp[j]$ (ในการหุงครั้งจากต้องการ i จะเหลือต้องการ j จานโดย $i-k\leq j\leq i-1$)

แต่เนื่องจากข้อนี้กำหนด $N \leq 100000, K \leq 100000$ วิธีนี้ที่ใช้เวลา $\mathcal{O}(NK)$ จึงช้าเกินไป เราจึงต้องใช้วิธีที่เร็วกว่านี้

เราสามารถกำหนด $C[i] = \Sigma_{j=0}^i dp[j]$ ซึ่งจะทำให้สามารถคำนวณ $dp[i] = C[i-1]$ เมื่อ $i-k-1 < 0$ และ $dp[i] = C[i-1] - C[i-k-1]$ เมื่อ $i-k-1 \geq 0$ ในเวลา $\mathcal{O}(1)$ ($C[i]$ สามารถคำนวณเป็น $C[i]=C[i-1] + dp[i]$ ในเวลา $\mathcal{O}(1)$ เช่นกัน)

ดังนั้นจึงต้องใช้เวลาเพียง $\mathcal{O}(1)$ สำหรับการคำนวณ $C[i]$ และ $dp[i]$ สำหรับแต่ละ $i \leq N$ ทั้งหมดจึงใช้เวลา $\mathcal{O}(N)$

โค้ดประกอบคำอธิบาย

PROGRAMMING.IN.TH

โปรแกรมมิ่งอินทีเอช ศูนย์รวมของโจทย์และเนื้อหาสำหรับ การเขียน
โปรแกรมเพื่อการแข่งขัน และวิทยาการคอมพิวเตอร์

ค้นหาโจทย์