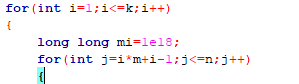
Destruction

ข้อนี้เราจะทำdp[k][n]นะโดยนิยามของdp[i][j]=มาถึงตัวที่iโดยต่อยไปjครั้งได้มูลค่าน้อยสุดเท่าไรแต่ว่าแต่ละครั้งที่ต่อยต้องมีความยาวติดต่อกัน>=mซึ่งถ้าอ่านโจทย์ดูดีๆมันจะคือข้อrose buyแบบซื้อkรอบทีนี้ถ้าเราไม่รู้อะไรเลยเราจะลองมาตั้งสมการdpแบบbrute forceก่อน

Dp[i][j]=qs[j]-qs[x]+dp[i-1][j-x-1] j-x>=m

ซึ่งเมื่อเราได้สมการแบบนี้มาที่เราต้องทำแบบbrute forceก็คือวนหาค่าx(j-mลงไปที่ค่ามากสุด)ที่ดีที่สุดใช่มะมันก็จะกลายเป็นo(n^2k)

แต่ว่าถ้าเราสังเกตดีๆค่าxนั้นจะน้อยกว่าjเสมอเราจึงสามารถเก็บมันมาก่อนได้โดยการเก็บใส่ตัวแปรซักตัวสมมติให้ตัวแปรนั้นชื่อmiละกันmi=-qs[j-m]+dp[i-1][j-m-1]ก็คือเราทำตามสมการข้างบนเลยอะ แต่อย่าลืมว่าข้อนี้เวลาต่อยสามารถต่อยโดยเว้นระยะห่างกี่ช่องก็ได้แต่ต้องห้ามติดกันเพราะฉะนั้นเราเลยต้องเทียบdp[i][j]=min(dp[i][j],dp[i][j-1])คือเลือกว่าจะต่อยดีมั้ยหรือเอาอันก่อนหน้าที่ดีกว่าเรา ข้อนี้น่าจะดักmemoryด้วยเราเลยต้องเปลี่ยนจากdp[k][n]กลายเป็นdp[2][n] มาดูการเขียนโค้ดนะ



ลูปiคือวนว่าเราตจะต่อยiครั้งส่วนลูปjวนจุดจบที่จะต่อยในแต่ละครั้ง ที่jต้องเริ่มจากi\*m+i-1นั้นเพราะว่าเราต้องต่อยiครั้งแต่ละครั้งยาวอย่างน้อยmและทุกครั้งต้องห้ามติดกันถ้าเราต่อยiครั้งมันเลยมีช่องว่างi-1อันพอเอามารวมกันมันเลยเป็นi\*m+i-1



ที่เราเทียบmaxของ0,j-m-1ในบรรทัดแรกเพราะว่าถ้าi=1แล้วj-m-1ติดลบแล้วก่อนเราจะเทียบminระกว่างdp[i%2][j]

กะdp[i%2][j-1]เราก็ต้องดูด้วยว่าjมันคือจุดเริ่มรึป่าวถ้าjคือจุดเริ่มแสดงว่าdpช่องก่อนหน้าไม่มีจริงสุดท้ายเค้าให้ตอบมูลค่าที่หลงเหลือจากการต่อยkครั้งนั่นคือqs[n]-dp[k%2][n]