

B05. 1092. 배.

① 각 크레인 제한 무게 : limit = [...]

② 각 박스 무게 : weight = [...]

cnt = 0

visited = [0] * M 0 → 안 옮김 1 → 옮김

while weight :

 for i in range(len(limit)):

 for j in range(len(weight)):

 if limit[i] >= weight[j]

 and visited[j] == 0 :

 visited[j] = 1

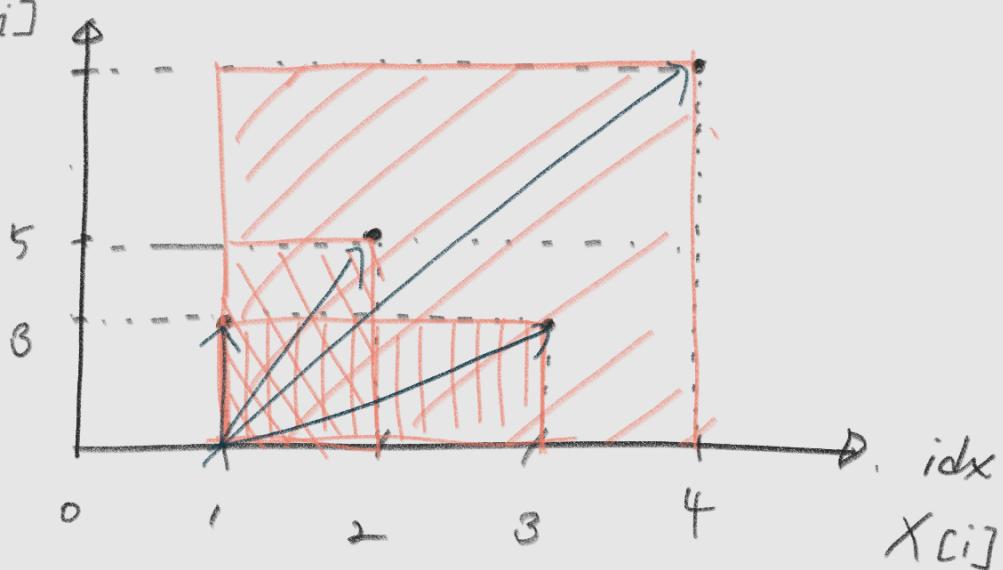
 break

cnt += 1

B05. 2141. 우체국

(인원) $A[i]$

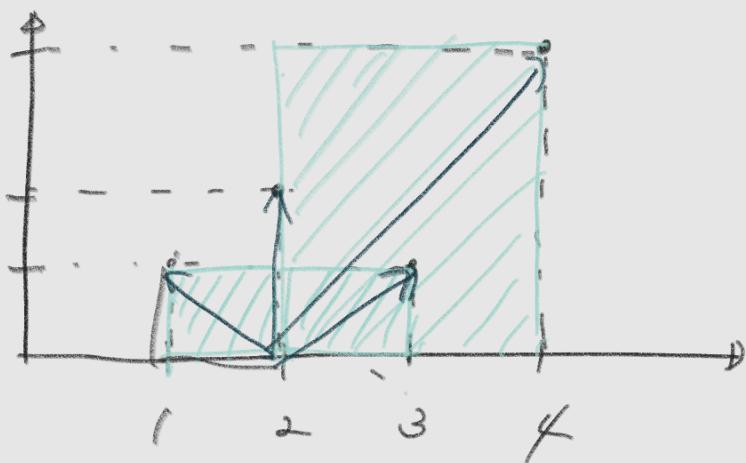
if 1이기준.



(직사각형 넓이)의 합.

(거4)

if 2가기준



조회평균?

$$X = \frac{a^2 + b^2 + \dots + n^2}{n}$$

8

11

(1, 3)	(2, 5)		(4, 0)	(10, 1)
1	0	5	30	9
2	3	0	20	8
3	6	5	10	7
4	9	10	0	6 ✓
5	12	15	10	5

18

4	0
30	
20	
10	
0	
10	
20	

10. 11

110

99

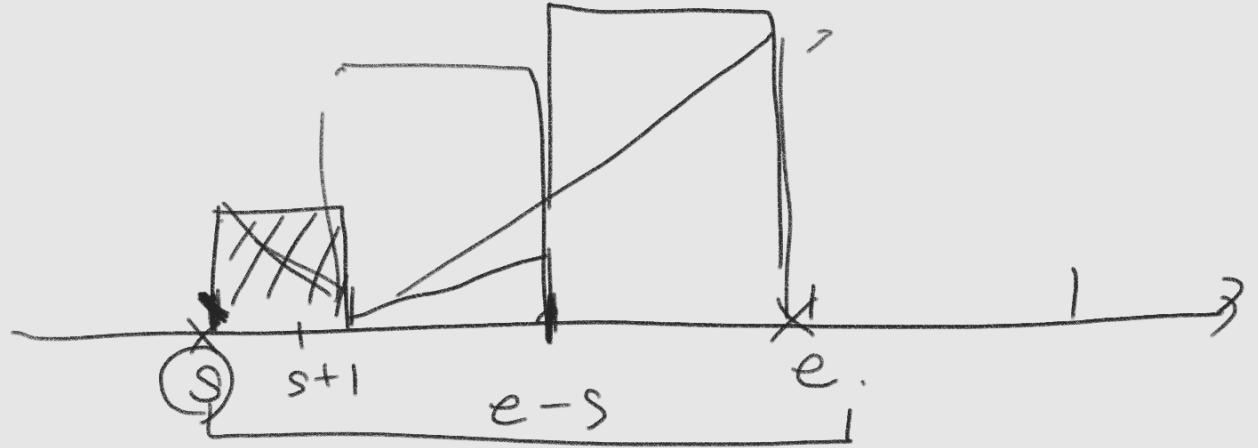
88

11 : 96.

66 = 103

55 = 110

$$\begin{array}{r} 87 \\ \times 9 \\ \hline 96 \end{array}$$



① s 를 기준으로 만들기?

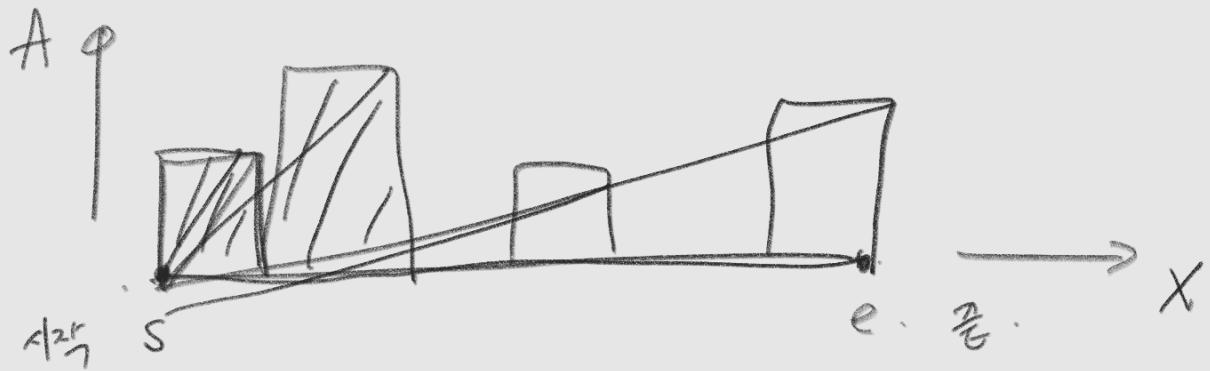
front, now, back, total
인구수

for i in range(s, e) :

total은 front 만큼 전체 증가, back 만큼 감소.

, 만일 내가 있는 지점이 $X[i]$ 라면

$$\left[\begin{array}{l} \text{now} = A[i] \\ \text{front} + = \text{now} \\ \text{back} - = \text{now} \end{array} \right]$$



Var.

total : 각각의 넓이 합

front : 앞쪽 인구 합

now : 지금 있는 인구

back : 뒷쪽 인구 합

nextX, nextA = info.pop(0)

for i in range(s, e+1) :

 total += front

 total -= back

if nextX == i :

 front += nextA

 back -= nextA

nextX, nextA = info.pop(0)

BOJ. 13975. 파일 합치기 3.

1) 문제.

파일 2개씩 합침 → 책 한권 만들려고 함

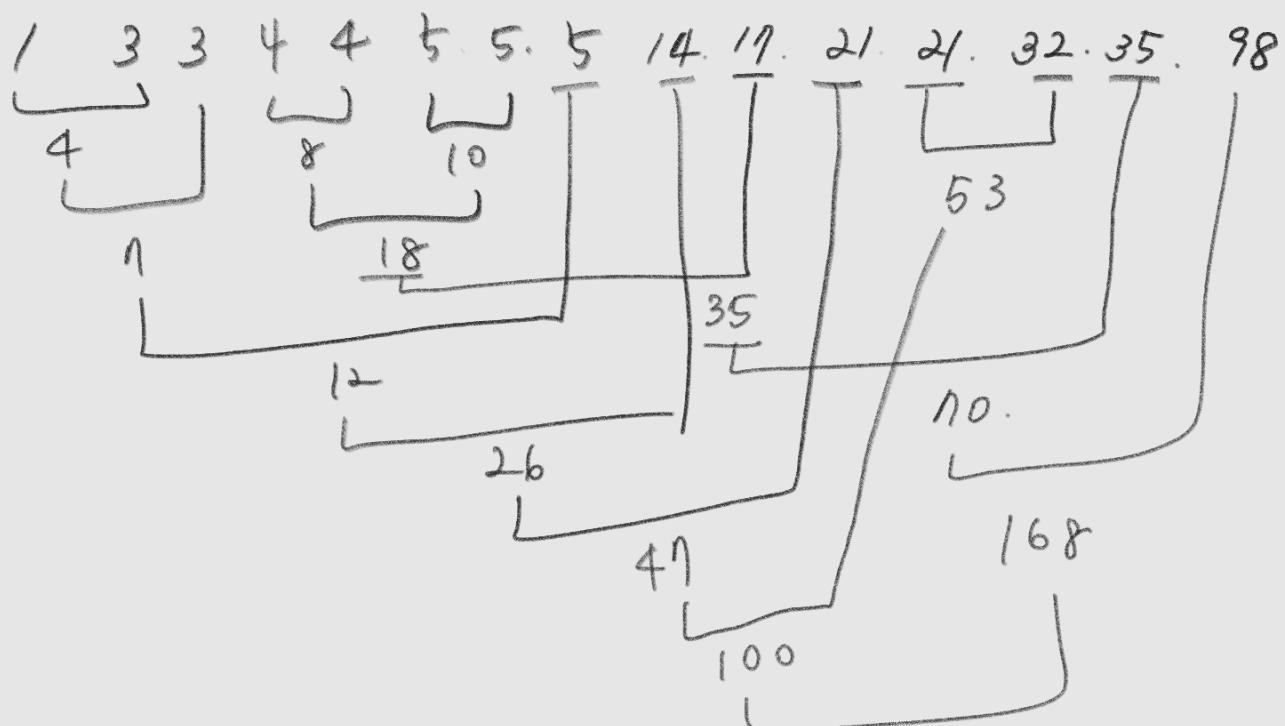
· 각각의 크기의 합만큼 비용 소모

· 최소비용은?

2) 풀이:

가장 작은거 2개씩 더해나가면 됨.

(Python은 최소 heap 이므로 heap을 사용하면 간편.)



아이디어 : ① 작은것 찾어서 ② 뽑고 ③ 더하고 ④ 넣기 ⑤ 반복.

B0J. 2812. 크게 만들기. G3

1) 문제 : N자리 숫자 주어짐.

· K개 자면서 만들 수 있는 가장 큰 수?

→ 제한 조건.

$$N \leq 500,000 \quad \text{자연수}$$

3) 풀이

· origin = ['1', '9', '2', '4']

· result = [결과 받을 리스트].

· cnt = K.

$i = 0$

$i = 0$

result = ['1'] cnt = 2 < '1', '9', '2', '4' >

$i = 2$

$i = 1$

result = ['9'] cnt = 1 < '1', '9', '2', '4' >

$i = 3$

$i = 2$

result = ['9'] cnt = 1 < '1', '9', '2', '4' >

$i = 4$

$i = 3$

result = ['9', '4'], cnt = 0. < '1', '9', '2', '4' >

BoJ. 11/15. 카드 정렬하기. G 4.

→ 파일 헉정기로 같은 문제.