

D : スキャナー

@Yazaten

問題概要

スキャン時間 T の異なる N 枚の紙をスキャンする。
3台のスキャナーを用いてスキャンしたとき
全ての紙のスキャンにかかる時間の最小を求めよ。

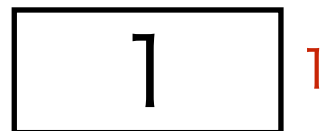
例 : $N=4$, $T=\{1,1,1,1\}$

スキャナー A



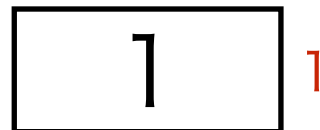
.....

スキャナー B



.....

スキャナー C



Answer : 2

問題概要

スキャン時間 T の異なる N 枚の紙をスキャンする。
3台のスキャナーを用いてスキャンしたとき
全ての紙のスキャンにかかる時間の最小を求めよ。

例 : $N=9$, $T=\{15,20,27,4,10,7,34,30,36\}$

スキャナー A

4	27	30
---	----	----

61

.....

スキャナー B

7	20	34
---	----	----

61

.....

スキャナー C

10	15	36
----	----	----

61

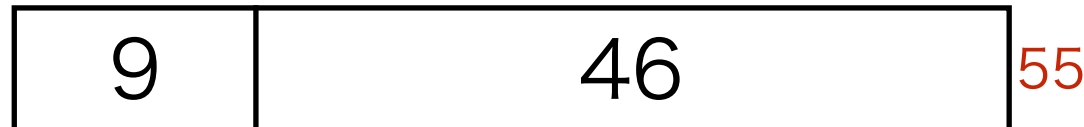
Answer : 61

問題概要

スキャン時間 T の異なる N 枚の紙をスキャンする。
3台のスキャナーを用いてスキャンしたとき
全ての紙のスキャンにかかる時間の最小を求めよ。

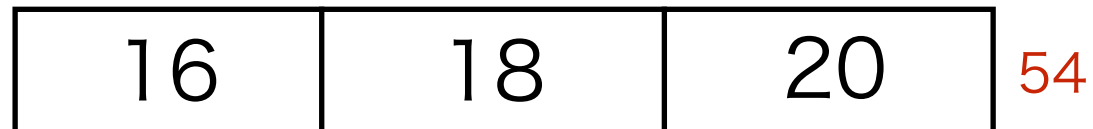
例 : $N=6$, $T=\{20,18,46,16,9,48\}$

スキャナー A



.....

スキャナー B



.....

スキャナー C



Answer : 55

考察

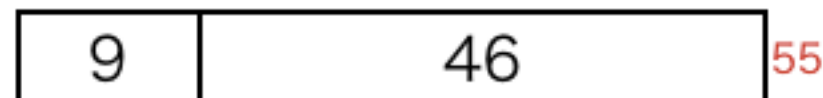
少し考えると、さまざまな貪欲法が思い浮かぶ
(空いたスキャナにどんどん紙をいれる とか…)

しかし

後続く紙によって、意外な割り当て方が最短になる
ケース(sample3など)があるので難しい

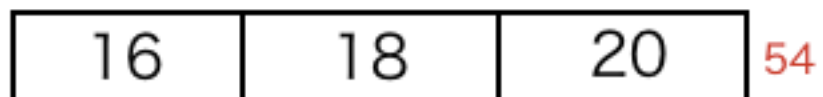
例 : $N=6$, $T=\{20,18,46,16,9,48\}$

スキャナー A



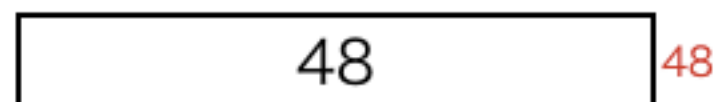
55

.....
スキャナー B



54

.....
スキャナー C



48

Answer : 55

考察

そこで

愚直に全探索

$$O(3^N)$$

各スキャナーが使用可能になる
時間を状態にしてDP

$$\frac{O(N^4 * (\sum T)^3)}{50^7 \doteq 78 * 10^{10}}$$

これではTLEする

(空間計算量も同様のオーダーになるのでMLEもする)

制約が小さい場合の解法

この問題の解法と先ほどのDPは考え方が近い
ため
ここでは先に「TLEするDP解法」についての解説を行う

スキャナーAを時間 j まで
スキャナーBを時間 k まで
スキャナーCを時間 l まで } 動かした時、

0~ i 番目の紙のスキャンを完了することができるか？

といった小問題を解くことで、この問題の解答が求まる

制約が小さい場合の解法

① 状態の持ち方

$dp[i][j][k][l]$

= スキャナA,B,Cをそれぞれ、ちょうど時間 j, k, l まで動作させたとき
 i 番目までの紙全てをスキャンすることはできるか？

(dp 配列の初期値は `false`)

制約が小さい場合の解法

② 遷移の仕方

$$\begin{aligned} \text{dp}[i][j][k][l] \mid = & \text{dp}[i-1][j-T[i]][k][l] \\ & \mid \mid \text{dp}[i-1][j][k-T[i]][l] \\ & \mid \mid \text{dp}[i-1][j][k][l-T[i]] \end{aligned}$$

右辺の $\text{dp}[i-1][j-T[i]][k][l]$ のいずれかが true のとき、
 $\text{dp}[i-1][j][k-T[i]][l]$
 $\text{dp}[i-1][j][k][l-T[i]]$

左辺($\text{dp}[i][j][k][l]$)は true に書き換えられる

※ $T[i]$ は*i*番目(1-index)の紙をスキャンするのにかかる時間

制約が小さい場合の解法

③ 最小値の探し方

「 $j+k+l == \Sigma T$ 」かつ「 $dp[n+1][j][k][l]==true$ 」 となるような j, k, l の組が最小値の候補となる。

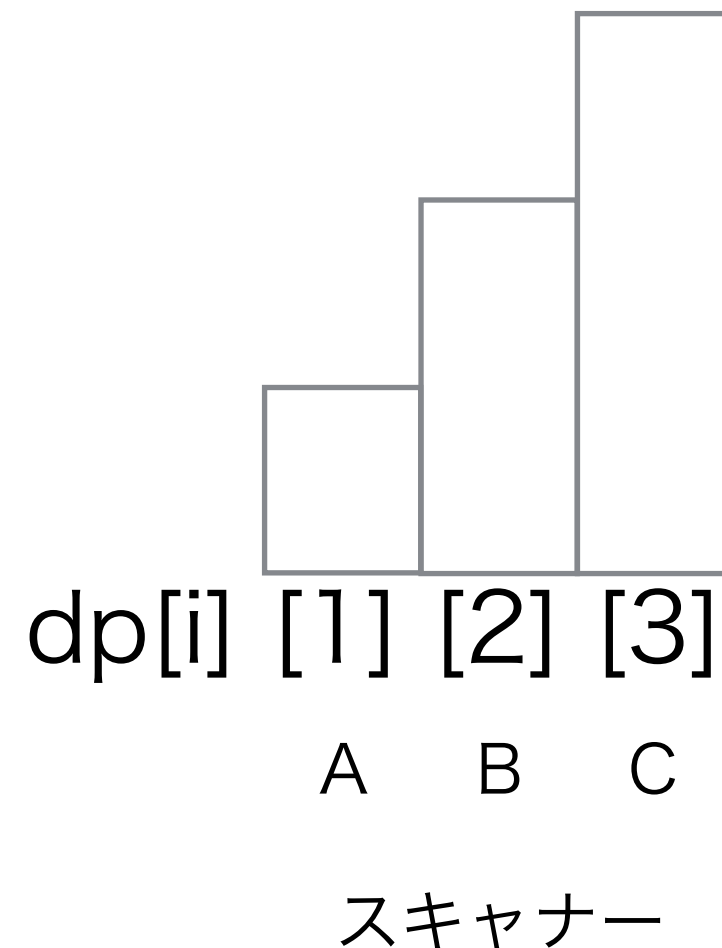
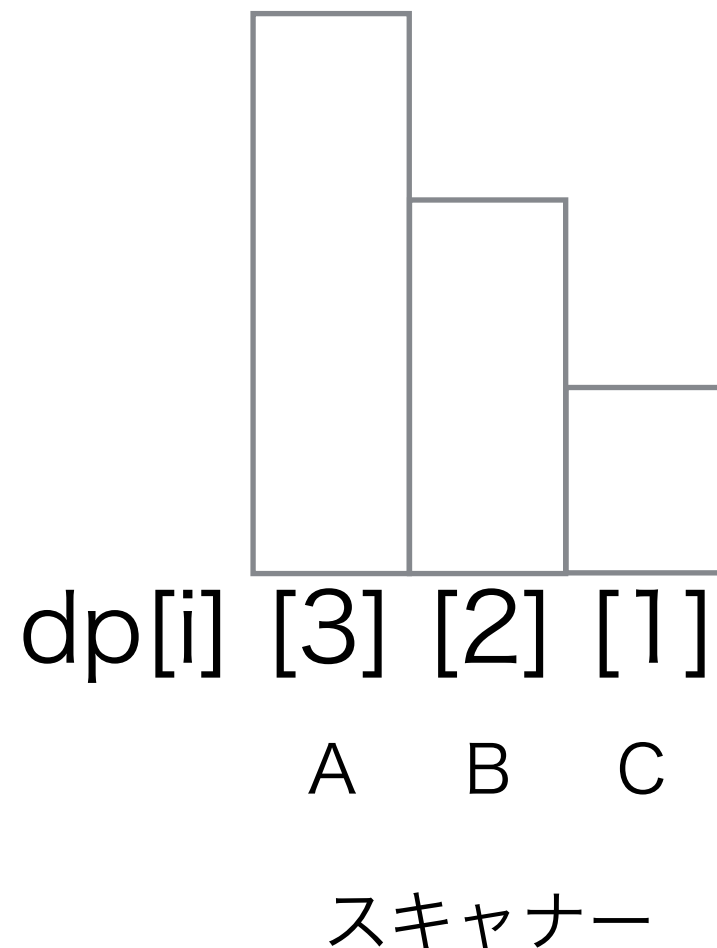
任意の j, k, l の組において、そのときにスキャンが終了した時間は $\min(j, k, l)$ となる。

つまり、全ての最小値の候補に対して $\min(j, k, l)$ の値を求め、その最小値がこの問題の答えとなる。

解法

実は今説明したDPには無駄がある

それぞれのスキャナA,B,Cに違いはないため、
以下の2つの図に示す状態は等価である



解法

同様に、以下の6個の状態の例は等価である

$dp[i][3][2][1]$, $dp[i][2][3][1]$, $dp[i][1][3][2]$
 $dp[i][3][1][2]$, $dp[i][2][1][3]$, $dp[i][1][2][3]$

このことから、状態として3つの”値”だけ持っておけばよいことがわかる

解法

また、 i 番目までの紙をスキャンするための時間がわかっていれば、スキャナーAとスキャナーBの時間からスキャナーCの時間を復元することができる

このことから、状態として2つの”値”だけ持っておけばよいことがわかる

解法

`bool dp[51][2500][2500]`というように状態を持ちうまく遷移させれば、高速かつ空間計算量を節約して解くことができる

AOJは早いので上の解法で通るが、ループ回数は約 3×10^8 と結構多くなる

各スキャナは高々 $2500/3 + 50$ 程度までしか稼働しない事に気づけば `dp[51][900][900]` とできるので、ループ回数が約 3×10^7 になって余裕を持って通る

実装上の注意など

メモリを結構使うので、配列を大きめにとっているとMLEしてしまう場合があるので気をつける

ジャッジ解

T.M : 29行

tubo28 : 68行

Yazaten : 42行

総評

AC/Submit : 49.12 %

FA-onsite : chipstar 31 min

FA-online : anta 11 min