Mugurel

Autor: stud. Ignat Alex-Matei, Universitatea Babeș-Bolyai, Cluj-Napoca

Problema se împarte de fapt în două subprobleme independente:

- aflarea numărului de kilograme minim pentru cutii și crearea șirului de cutii
- împărțirea șirului de cutii în două loturi care au suma discrepanțelor minimă

Mai departe, vom aborda separat soluțiile pentru cele două subprobleme.

Numărul de kilograme al cutiilor și șirul de cutii

Observație 1. Din restricțiile problemei cu privire la cutiile mixte, se observă că suma matricilor grămezilor de portocale și banane formează matricea mixtă pe care ar trebui să lucrăm, dacă alegem să împachetăm mixt fructele.

Observație 2. Problema impune următoarea strategie: găsirea numărului minim de kilograme pentru cutiile de banane, apoi pentru cele de portocale, apoi pentru cele mixte, iar varianta de împachetare se va alege conform:

$$min(\mathrm{c}_{port} \cdot A + \mathrm{c}_{ban} \cdot B, \mathrm{c}_{mixt} \cdot C)$$

unde c_{port}, c_{ban}, c_{mixt} reprezintă numărul minim de kilograme al cutiilor pentru fiecare tip în parte.

Deci, problema ***găsirii numărului de kilograme*** se rezolvă prima dată independent pentru portocale, banane și mixt, apoi se combină soluțiile. În continuare, vom nota cu $fructe$ una dintre cele trei matrice (iar ideile explicate în continuare se aplică de 3 ori la fel, pe rând, pe fiecare matrice).

În funcție de care dintre valori este cea mai mică, vom alege să împachetăm fructele separat sau împreună.

Observație 3. Dacă știm numărul de kilograme al unei cutii, împachetarea se poate realiza foarte ușor. Numim *simulare* următoarea metodă de împachetare în cutii cu un număr cunoscut de kilograme maxim: la fiecare moment, avem o singură cutie deschisă (pentru tipul respectiv de fructe). Dacă grămada încape în cutie, o adăugăm (dacă nu depășim numărul maxim de kilograme). Altfel, cutia se închide și se deschide una nouă în care punem grămada curentă. La finalul zilei, după ce am parcurs toate grămezile, închidem cutia care a rămas deschisă. Prin această strategie se formează șirul de cutii dacă cunoaștem numărul de kilograme al cutiilor. Mai departe, vom vedea cum putem să aflăm acest număr de kilograme.

Subtask 1. Avem $K = N$ cutii.

Știm că toate fructele trebuie împachetate, iar cutiile nu pot să conțină fructe din zile diferite, ceea ce înseamnă că se alocă câte o cutie în fiecare zi. În plus, din moment ce toate grămezile trebuie împachetate, toate grămezile dintr-o zi anume vor fi puse într-o singură cutie. Așadar, numărul de kilograme ale cutiilor de un anumit tip va fi egal cu suma maximă a unei linii din matricea de fructe. Fiecare linie determină o cutie, deci pentru a crea șirul, calculăm sumele liniilor și punem aceste sume într-un alt șir.

Complexitate: $O(N \cdot M)$

Subtask 2. Avem $K = N + 1$ cutii.

Din primul subtask, știm că dacă avem N cutii, în fiecare zi se alocă o singură cutie. La acest subtask, mai avem o cutie în plus pe care o putem folosi. Din moment ce vrem să minimizăm cutia cu număr maxim de kilograme, această ultimă cutie trebuie alocată liniei (zilei) cu număr maxim de kilograme (vrem să împărțim cutia maximă în alte două cutii, încercând astfel minimizarea). Presupunem că linia i este cea care determină cutia maximă. Deci, căutăm un moment de timp (o coloană) j astfel încât să minimizăm maximul dintre suma elementelor de la coloana 1 la coloana j și suma elementelor de la coloana $j+1$ la coloana M, pe linia i. Acest lucru se poate face cu ajutorul unor sume parțiale pe linia i, sau calculând suma totală și scăzând de fiecare dată elementul de pe poziția j ($j \to 1, M$), sau se acceptă și luarea pe rând a coloanelor și calcularea sumei din stânga și sumei din dreapta de fiecare dată.

Complexitate: $O(N \cdot M + N^2)$ sau $O(N \cdot M + N)$

Subtask 3. Toate grămezile au același număr de kilograme.

Generalizând ideea de la *Subtask 2*, de fiecare dată când vrem să alocăm o nouă cutie, aceasta ar trebui alocată cutiei cu număr maxim de kilograme. Considerăm momentan $K = N$ cutii. Din moment ce toate grămezile sunt identice și toate zilele au același număr de grămezi, se observă că nu contează dacă $K = N$ sau $K \leq 2 \cdot N - 1$, pentru că ar trebui să alocăm cutii în plus pentru fiecare zi pentru a minimiza cu adevărat cutia maximă. De aici, se poate concluziona că în fiecare zi trebuie să alocăm același număr de cutii (putem aloca și mai multe într-o anumită zi, dar fără niciun efect real în minimizarea kilogramelor cutiilor, din moment ce toate zilele arată identic). Așadar, numărul de cutii alocate în fiecare zi va fi:

$$[\frac{K}{N}]$$

Totodată, vrem ca aceste cutii din fiecare zi să aibă un număr cât mai apropiat de kilograme una de cealaltă, deci vom încerca să punem un număr de grămezi cât mai egal în fiecare cutie. Notăm cu X numărul de cutii alocate în fiecare zi. Numărul de grămezi al cutiei maxime va fi egal cu:

$$[\frac{M}{X}] + Y$$

unde $Y = 0$ dacă $M \ mod \ X = 0$, sau $Y = 1$ în caz contrar.

Complexitate: $O(N \cdot M)$

Subtask 4. Suma grămezilor din oricare zi este ≤ 1000, iar M și N sunt ≤ 100.

Din moment ce suma grămezilor din fiecare zi este maxim 1000, înseamnă că nici capacitatea cutiilor nu va fi mai mare de 1000. Astfel, putem să trecem prin toate capacitățile (de la 1 la 1000) și să vedem dacă putem să facem o împachetare, având cutiile cu acel număr de kilograme, folosind bineînțeles maxim K cutii (lucru care se face printr-o *simulare*). Prima capacitate care respectă acest lucru este chiar cea minimă pe care noi o căutăm.

Complexitate: $O(N \cdot M \cdot max_i(\sum_{j=1}^{M} fructe[i][j]))$

Subtask 5, 6. Vom aborda o tactică similară celei de la Subtask 4. Vom face în continuare o *simulare* pentru fiecare capacitate. Cum am putea însă să căutăm mai optim capacitatea fără să trecem prin toate cele posibile? Ideea va fi să facem o ***căutare binară pe răspuns***. Reținem:

$$st = min(fructe[i][j])$$

$$dr = max_i(\sum_{j=1}^{M} fructe[i][j])$$

unde i și j reprezintă o linie și o coloană a matricei cu fructe. La un pas al căutării binare, vom avea:

$$mij = (st + dr)/2$$

Facem o *simulare*, unde considerăm capacitatea cutiilor $= mij$. În urma rezultatului, concluzionăm:

- Dacă reușim să facem împachetarea și am folosit maxim K cutii, atunci este clar că împachetarea se poate face și dacă am crește capacitatea cutiilor. Deci nu are rost să verificăm capacitățile strict mai mari.

$$dr \leftarrow mij$$

- Dacă nu reușim să facem împachetarea, atunci este clar că împachetarea nu se poate face nici dacă am scădea capacitatea cutiilor. Deci nu are rost să verificăm capacitățile mai mici sau egale.

$$st \leftarrow mij + 1$$

Ne oprim în momentul în care $st = dr$, în oricare din variabile fiind de fapt numărul de kilograme căutat.

Complexitate: $O(N \cdot M \cdot log(\ max_i(\sum_{j=1}^{M} fructe[i][j])))$

Împărțirea șirului în două loturi cu suma discrepanțelor minimă

În continuare, trebuie să găsim o împărțire a șirului de cutii, astfel încât să minimizăm suma celor două discrepanțe a celor două loturi. Reamintim că o discrepanță a unui lot este diferența dintre cutia maximă și cutia minimă (care pot să coincidă - un lot poate să conțină și o singură cutie).

Mai departe, notăm cu S șirul de cutii și Q numărul de cutii.

Subtask 1, 2, 4, 5. În principiu, numărul de cutii care se creează este $< 10^4$.

Putem face un ***brute force*** pentru a afla discrepanțele. Parcurgem toate pozițiile de la 1 la $Q-1$. Notăm o poziție curentă cu i. Pentru fiecare poziție, parcurgem și reținem cutia minimă și cutia maximă din subșirul de la 1 la i și apoi din subșirul de la $i+1$ la Q. Discrepanța minimă va fi:

$$min_i(max_{S[1 \to i]} - min_{S[1 \to i]} + max_{S[i+1 \to Q]} - min_{S[i+1 \to Q]})$$

Complexitate: $O(Q^2)$

Subtask 3, 6. $10^4 \leq Q \leq 10^6$

Pentru a rezolva mai eficient, vom crea următoarele două șiruri:

$$\begin{aligned}
min_{st}[i] &= \text{cutia minimă din subșirul } [1 \to i] \\
max_{st}[i] &= \text{cutia maximă din subșirul } [1 \to i] \\
min_{dr}[i] &= \text{cutia minimă din subșirul } [i \to Q] \\
max_{st}[i] &= \text{cutia maximă din subșirul } [i \to Q]
\end{aligned}$$

Cele două șiruri din stânga se parcurg de la 1 la Q, iar cele două din dreapta se parcurg de la Q la 1, și se formează prin următoarea recurență:

$$min_{st}[i] = \begin{cases} S[i], & \text{dacă } i = 1 \\ min(min_{st}[i-1], S[i]), & \text{altfel} \end{cases}$$

$$\max_{st}[i] = \begin{cases} S[i], & \text{dacă } i = 1 \\ max(max_{st}[i-1], S[i]), & \text{altfel} \end{cases}$$

$$\min_{dr}[i] = \begin{cases} S[i], & \text{dacă } i = Q \\ min(min_{dr}[i+1], S[i]), & \text{altfel} \end{cases}$$

$$\max_{dr}[i] = \begin{cases} S[i], & \text{dacă } i = Q \\ max(max_{dr}[i+1], S[i]), & \text{altfel} \end{cases}$$

Luăm fiecare poziție i (de la 1 la $Q-1$) în parte, iar suma minimă a discrepanțelor se află după următoarea formulă:

$$min_i(max_{st}[i] - min_{st}[i] + max_{dr}[i+1] - min_{dr}[i+1])$$

Precalcularea șirurilor se face în complexitate $O(Q)$, la fel și aflarea sumei minime dintre discrepanțe, obținând complexitatea finală $O(Q)$.

Notă

Complexitatea unui *subtask* este dată de suma celor două complexități ale subproblemelor în parte.

Complexitățile sunt de timp, memoria nu ar trebui să creeze probleme.

La *subtask 5 și 6*, în căutarea binară pentru aflarea numărului minim de kilograme, *st* și *dr* pot fi inițializate și cu valorile minime și maxime posibile indiferent de caz, adică $st \leftarrow 1$, respectiv $dr \leftarrow 10^9$.

Exercițiu: Cum putem face împărțirea eficient dacă ar trebui împărțit șirul în 4 loturi cu suma discrepanțelor minimă? Hint: Complexitate $O(Q^2)$