

Detecting Molecules

פטר (Petr) עובד בחברה שבנתה מכונה לזיהוי מולקולות. לכל מולקולה יש משקל שהוא מספר שלם חיובי. למכונה יש טווח זיהוי $[l, u]$, כאשר l ו- u הם מספרים שלמים חיוביים. המכונה יכולה לזהות קבוצת מולקולות אם ורק אם הקבוצה מכילה תת קבוצה של מולקולות עם משקל כולל שנמצא בטווח הזיהוי של המכונה.

פורמלית, נתונות n מולקולות עם משקלים w_0, \dots, w_{n-1} . הזיהוי יצליח אם קיימת קבוצה של אינדקסים שונים

$$I = \{i_1, \dots, i_m\} \text{ כך שמתקיים } l \leq w_{i_1} + \dots + w_{i_m} \leq u$$

מהגדרת המכונה, מובטח שההפרש בין l ו- u גדול או שווה להפרש המשקלים בין המולקולה הכבדה ביותר לקלה ביותר. באופן פורמלי, מתקיים $u - l \geq w_{max} - w_{min}$, כאשר נגדיר $w_{max} = \max(w_0, \dots, w_{n-1})$ וגם

$$w_{min} = \min(w_0, \dots, w_{n-1})$$

משימתכם היא לכתוב תוכנית שמוצאת תת קבוצה כלשהי של מולקולות עם משקל כולל שנמצא בטווח הזיהוי, או קובעת שאין תת קבוצה כזו.

פרטי מימוש

עליכם לממש פונקציה אחת (שיטה):

• `int[] solve(int l, int u, int[] w)`

○ l ו- u : הקצוות של טווח הזיהוי.

○ w : משקלי המולקולות.

○ אם תת הקבוצה הנדרשת קיימת, הפונקציה צריכה להחזיר מערך של אינדקסים של מולקולות

אשר יוצרות תת קבוצה כזו. אם קיימות מספר תשובות נכונות, החזירו אחת מהן.

○ אם תת קבוצה כזו לא קיימת, הפונקציה צריכה להחזיר מערך ריק.

עבור שפת C חתימת הפונקציה שונה מעט:

• `int solve(int l, int u, int[] w, int n, int[] result)`

○ n : מספר האיברים ב- w (כלומר, מספר המולקולות).

○ שאר הפרמטרים זהים למתואר לעיל.

○ במקום להחזיר מערך של m אינדקסים (כמתואר לעיל), הפונקציה צריכה לכתוב את האינדקסים

ל- m התאים הראשונים של המערך `result` ואז להחזיר את `m`.

○ אם תת הקבוצה הנדרשת לא קיימת, הפונקציה צריכה לא לכתוב כלום למערך `result` והיא

צריכה להחזיר את הערך 0.

לתוכניתכם מותר לכתוב את האינדקסים למערך המוחזר (או המערך `result` בשפת C) בכל סדר שהוא.

אנא השתמשו בקבצי ה-`template` לפרטי המימוש בשפת התכנות שלכם.

דוגמאות

דוגמה 1

$\text{solve}(15, 17, [6, 8, 8, 7])$

בדוגמה זו יש לנו ארבע מולקולות עם משקלים 6, 8, 8 ו-7. המכונה יכולה לזהות תת קבוצות של מולקולות עם סך משקלים בין 15 ל-17, כולל. שימו לב שמתקיים $17 - 15 \geq 8 - 6$. המשקל הכולל של המולקולות 1 ו-3 הוא $w_1 + w_3 = 8 + 7 = 15$, לכן הפונקציה יכולה להחזיר $[1, 3]$. תשובות נכונות נוספות הן $[1, 2]$ (כי $w_1 + w_2 = 8 + 8 = 16$) וגם $[2, 3]$ (כי $w_2 + w_3 = 8 + 7 = 15$).

דוגמה 2

$\text{solve}(14, 15, [5, 5, 6, 6])$

בדוגמה זו יש לנו ארבע מולקולות עם משקלים 5, 5, 6 ו-6, ואנחנו מחפשים תת קבוצה שלהן עם סך משקלים בין 14 ל-15, כולל. שוב, שימו לב שמתקיים $15 - 14 \geq 6 - 5$. לא קיימת תת קבוצה של מולקולות עם סך משקלים בין 14 ל-15, לכן הפונקציה צריכה להחזיר מערך ריק.

דוגמה 3

$\text{solve}(10, 20, [15, 17, 16, 18])$

בדוגמה זו יש לנו ארבע מולקולות עם משקלים 15, 17, 16 ו-18, ואנחנו מחפשים תת קבוצה שלהן עם סך משקלים בין 10 ל-20, כולל. שוב, שימו לב שמתקיים $20 - 10 \geq 18 - 15$. לכל תת קבוצה שמכילה איבר בודד יש סך משקלים בין 10 ל-20, לכן התשובות הנכונות האפשריות הן: $[0]$, $[1]$, $[2]$ ו- $[3]$.

תת משימות

1. (9 נקודות): $1 \leq n \leq 100$, $1 \leq w_i \leq 100$, $1 \leq u, l \leq 1000$, וכל ערכי w_i שווים.
2. (10 נקודות): $1 \leq n \leq 100$, $1 \leq w_i, u, l \leq 1000$, וגם $\max(w_0, \dots, w_{n-1}) - \min(w_0, \dots, w_{n-1}) \leq 1$.
3. (12 נקודות): $1 \leq n \leq 100$, וגם $1 \leq w_i, u, l \leq 1000$.
4. (15 נקודות): $1 \leq n \leq 10\,000$, וגם $1 \leq w_i, u, l \leq 10\,000$.
5. (23 נקודות): $1 \leq n \leq 10\,000$, וגם $1 \leq w_i, u, l \leq 500\,000$.
6. (31 נקודות): $1 \leq n \leq 200\,000$, וגם $1 \leq w_i, u, l < 2^{31}$.

גריידר לדוגמה

הגריידר קורא קלט בפורמט הבא:

- שורה ראשונה: שלושה מספרים שלמים, n ואחריו l ואחריו u .
- שורה שנייה: n מספרים שלמים: w_0, \dots, w_{n-1} .