

فیل‌های رقصنده

«فیل‌های رقصنده» یک نمایش (show) پریننده در پاتایا است که در آن N فیل روی یک خط (که به آن صحنه می‌گویند) می‌رقصند.

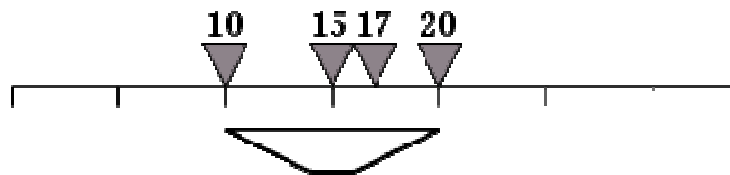
پس از سال‌ها آموزش، فیل‌های این نمایش قادر به انجام رقص‌های جالبی هستند. نمایش شامل تعدادی اجرا (act) است که در هر اجرا، دقیقاً یک فیل رقص زیبایی انجام می‌دهد و در این حین ممکن است مکان خود را عوض کند.

تهیه‌کنندگان نمایش می‌خواهند کتاب عکسی تولید کنند که عکس‌های کل نمایش را داشته باشد. بعد از هر اجرا، آن‌ها می‌خواهند عکس‌هایی از همه فیل‌ها، به همان شکلی که بینندگان می‌بینند، بگیرند.

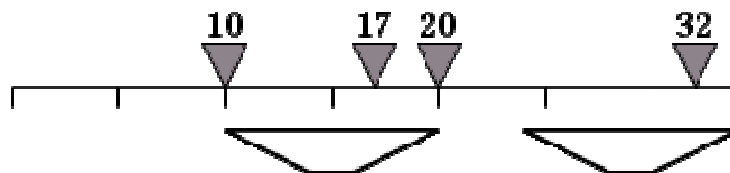
در هر زمانی از نمایش، تعدادی فیل ممکن است مکان یکسانی داشته باشند. در این صورت، آن‌ها پشت هم در همان مکان قرار می‌گیرند.

یک دوربین می‌تواند یک عکس از یک گروه از فیل‌ها بگیرد اگر و تنها اگر مکان آن گروه از فیل‌ها روی پاره‌خطی به طول L (شامل هر دو نقطه‌ی دو سر پاره‌خط) جا شود. از آن‌جا که فیل‌ها می‌توانند روی صحنه پخش شوند ممکن است به چندین دوربین برای ثبت هم‌زمان تصویر همه فیل‌ها نیاز داشته باشیم.

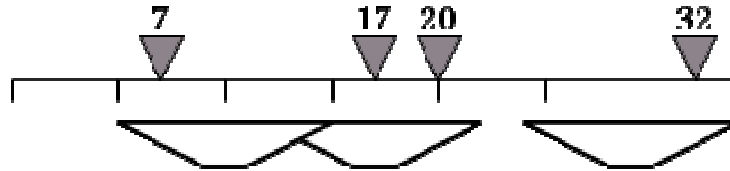
برای مثال، فرض کنید $L = 10$ و فیل‌ها در مکان‌های ۱۰، ۱۵، ۱۷ و ۲۰ صحنه قرار دارند. در این لحظه، یک دوربین (همان‌طور که در شکل زیر می‌بینید) می‌تواند تصویرشان را بگیرد. (فیل‌ها در شکل به صورت مثلث و دوربین به صورت مثلث وارون دیده می‌شوند).



در اجرای بعدی، فیل‌هایی که در مکان ۱۵ هست رقص کنان به مکان ۳۲ می‌رود. بعد از این اجرا، ما نیاز به حداقل ۲ دوربین برای ثبت تصویر داریم.



در اجرای بعدی، فیل‌هایی که در مکان ۱۰ است به مکان ۷ می‌رود. در این وضعیت جدید فیل‌ها، ما نیاز به ۳ دوربین برای عکس گرفتن از همه فیل‌ها داریم.



در این مسئله‌ی تعاملی (Interactive)، شما باید کمترین تعداد دوربین مورد نیاز برای تصویربرداری بعد از هر اجرا را معین کنید. توجه کنید که بین هر دو اجرای متوالی، تعداد دوربین‌های مورد نیاز ممکن است زیاد یا کم شود و یا تغییری نکند.

وظیفه شما

شما باید روال‌های زیر را پیاده‌سازی کنید.

• روال $\text{init}(N, L, X)$ که پارامترهای زیر را می‌گیرد:

○ N : تعداد فیل‌ها که با شماره‌های صفر تا $N - 1$ شماره‌گذاری شده‌اند.

○ L : طول پاره‌خطی است که با یک دوربین به تنهایی ضبط می‌شود. می‌توانید فرض کنید که

$0 \leq L \leq 1,000,000,000$ است.

○ X : یک آرایه‌ی یک‌بُعدی از اعداد صحیح که مکان آغازین فیل‌ها را مشخص می‌کند. برای $0 \leq i < N$ ،

مکان آغازین فیل i -ام برابر با $X[i]$ است. مکان‌های آغازین به صورت صعودی مرتب‌شده‌اند. یعنی

می‌توانید فرض کنید که $0 \leq X[0] \leq \dots \leq X[N - 1] \leq 1,000,000,000$ است. توجه داشته

باشید که در حین رقص، فیل‌ها ممکن است مکان‌شان را تغییر دهند.

این روال فقط یک بار (در ابتدای کار و پیش از همه‌ی فراخوانی‌های روال `update`) صدا زده خواهد شد و هیچ

مقداری را برنمی‌گرداند.

• روال $\text{update}(i, y)$ که پارامترهای زیر را می‌گیرد:

○ i : شماره‌ی فیلی که در اجرای فعلی حرکت می‌کند.

○ y : مکان فیل i بعد از اجرا. می‌توانید فرض کنید که $0 \leq y \leq 1,000,000,000$ است و y یک عدد صحیح

است.

این روال چندین بار فراخوانی می‌شود. هر فراخوانی معادل یک اجرا است (که بعد از اجراهای قبلی انجام می‌شود). هر

فراخوانی باید کمترین تعداد دوربین مورد نیاز برای تصویربرداری از همه‌ی فیل‌ها بعد از اجرای متناظر را برگرداند.

مثال نمونه

مثال اول

حالتی را در نظر بگیرید که $N = 4$ ، $L = 10$ و مکان آغازین فیلها به صورت زیر است:

$$X = \begin{matrix} 10 \\ 15 \\ 17 \\ 20 \end{matrix}$$

ابتدا، روال `init` شما با این پارامترها فراخوانی خواهد شد. پس از آن، روال `update` شما یکبار برای هر اجرا فراخوانی می‌شود. در مثال زیر، دنباله‌ای از فراخوانی‌ها و مقدار صحیح برگشتی آن‌ها آمده است:

| شماره اجرا | پارامترهای ارسالی | مقدار برگشتی |
|------------|----------------------------|--------------|
| ۱ | <code>update(2, 16)</code> | 1 |
| ۲ | <code>update(1, 25)</code> | 2 |
| ۳ | <code>update(3, 35)</code> | 2 |
| ۴ | <code>update(0, 38)</code> | 2 |
| ۵ | <code>update(2, 0)</code> | 3 |

زیرمسئله‌ها

زیرمسئله شماره یک (۱۰ امتیاز)

- دقیقاً $N = 2$ فیل وجود دارند.
- در ابتدا و بعد از هر اجرا، مکان فیلها متفاوت است.
- روال `update` شما حداکثر ۱۰۰ بار فراخوانی می‌شود.

زیرمسئله شماره چهار (۴۷ امتیاز)

- $1 \leq N \leq 70,000$
- در یک مکان ممکن است بیشتر از یک فیل قرار گیرد.
- روال `update` شما حداکثر ۷۰,۰۰۰ بار فراخوانی می‌شود.

زیرمسئله شماره دو (۱۶ امتیاز)

- $1 \leq N \leq 100$
- در ابتدا و بعد از هر اجرا، مکان فیلها متفاوت است.
- روال `update` شما حداکثر ۱۰۰ بار فراخوانی می‌شود.

زیرمسئله شماره پنج (۳ امتیاز)

- $1 \leq N \leq 150,000$
- در یک مکان ممکن است بیشتر از یک فیل قرار گیرد.
- روال `update` شما حداکثر ۱۵۰,۰۰۰ بار فراخوانی می‌شود.
- نکته مربوط به محدودیت زمانی CPU را در قسمت جزئیات پیاده‌سازی ببینید.

زیرمسئله شماره سه (۲۴ امتیاز)

- $1 \leq N \leq 50,000$
- در ابتدا و بعد از هر اجرا، مکان فیلها متفاوت است.
- روال `update` شما حداکثر ۵۰,۰۰۰ بار فراخوانی می‌شود.

جزئیات پیاده‌سازی

محدودیت‌ها

- محدودیت زمانی CPU: ۹ ثانیه
- توجه: collection template در کتابخانه استاندارد C++ (STL) ممکن است کند باشند. مخصوصاً، اگر از آن‌ها استفاده کنید، ممکن است نتوانید زیر مسئله ۵ را حل کنید.
- محدودیت حافظه: ۲۵۶ مگابایت
- توجه: هیچ محدودیت صریحی روی اندازه‌ی حافظه‌ی پشته (stack) استفاده شده وجود ندارد. میزان حافظه‌ی استفاده شده در پشته به‌عنوان بخشی از حافظه‌ی مصرفی محاسبه می‌شود.

واسط API

- فولدر پیاده‌سازی: elephants/
 - فایل‌هایی که توسط شرکت‌کننده می‌بایست پیاده‌سازی شوند: elephants.c یا elephants.cpp یا elephants.pas
 - واسط شرکت‌کننده: elephants.h یا elephants.pas
 - مصحح نمونه: grader.c یا grader.cpp یا grader.pas
 - ورودی مصحح نمونه: grader.in.1, grader.in.2 و ...
 - توجه: مصحح نمونه، ورودی را به فرمت زیر می‌خواند:
 - خط اول: مقادیر N ، L و در نهایت M ، که M تعداد اجراها در نمایش است.
 - خط‌های ۲ الی $N + 1$ ام: مکان‌های آغازین. یعنی خط $K + 2$ (برای $0 \leq K < N$) شامل مقدار $X[K]$ است.
 - خط‌های $N + 2$ تا $N + M + 1$ ام: اطلاعات مربوط به M اجرا. یعنی برای هر $1 \leq j \leq M$ ، خط $N + 1 + j$ شامل سه مقدار $[j]$ ، $[j]$ ، $[j]$ و $S[j]$ که با یک فاصله از هم جدا شده‌اند به این معنی که در j -امین اجرا، فیل شماره $[j]$ به مکان $[j]$ می‌رود و بعد از آن اجرا، $S[j]$ کم‌ترین تعداد دوربین مورد نیاز برای تصویربرداریست.
 - خروجی مورد انتظار برای ورودی مصحح نمونه: grader.expect.1, grader.expect.2 و ...
- برای این مسئله، هر کدام از این فایل‌ها دقیقاً باید شامل عبارت «Correct.» باشند.