



برنامه‌ی بینایی

شما می‌خواهید یک برنامه‌ی بینایی برای یک روبات پیاده‌سازی کنید. هر بار که دوربین روبات یک عکس می‌گیرد، آن را به صورت یک تصویر سیاه و سفید در حافظه‌اش ذخیره می‌کند. هر تصویر از $W \times H$ پیکسل تشکیل شده است که سطرها از ۰ تا H و ستون‌ها از ۰ تا W شماره‌گذاری شده‌اند. در هر تصویر دقیقاً دو پیکسل سیاه وجود دارد و بقیه‌ی پیکسل‌ها سفید هستند.

روبات می‌تواند هر تصویر را با برنامه‌ای که شامل دنباله‌ای از دستورات ابتدایی است پردازش کند. مقادیر H و W یک عدد صحیح مثبت K به شما داده می‌شود. هدف شما نوشتن یکتابع برای تولید برنامه‌ای برای روبات است طوری که برای هر تصویر، مشخص کند آیا **فاصله‌ی** بین دو پیکسل سیاه دقیقاً K هست یا خیر. فاصله‌ی بین پیکسلی که در سطر r_1 و ستون c_1 قرار دارد با پیکسلی که در سطر r_2 و ستون c_2 قرار دارد، برابر با $|r_2 - r_1| + |c_2 - c_1|$ تعریف می‌شود. در این رابطه، $|x|$ به معنی قدر مطلق x است، یعنی برای $0 \leq x \leq$ برابر با x و برای $x < 0$ برابر با $-x$ است.

در ادامه نحوه‌ی کارکرد روبات توضیح داده می‌شود.

حافظه‌ی روبات یک آرایه‌ی کافی بزرگ از خانه‌هایی است که از ۰ اندیس‌گذاری شده‌اند. هر خانه‌ی حافظه می‌تواند مقدار ۰ یا ۱ را ذخیره کند و مقدار آن خانه پس از ذخیره قابل تغییر نیست. تصویر به صورت سطر به سطر در خانه‌های با اندیس ۰ تا $H \times W - 1$ ذخیره شده است. اولین سطر تصویر در خانه‌های با اندیس ۰ تا $W - 1$ و آخرین سطر تصویر در خانه‌های $(H - 1) \times W - 1$ ذخیره شده است. به طور خاص، اگر پیکسلی که در سطر i و ستون j قرار دارد سیاه باشد، مقدار خانه‌ی $j \times W + i$ حافظه برابر با ۱ بوده و در غیر این صورت برابر با ۰ است.

برنامه‌ی روبات یک دنباله از **دستورات** است که با اعداد صحیح متوالی به ترتیب از ۰ شماره‌گذاری شده‌اند. وقتی این برنامه اجرا می‌شود، دستوارت یک به یک اجرا می‌شوند. هر دستور مقادیر مربوط به یک یا چند خانه (که این مقادیر **ورودی‌های** دستور نامیده می‌شوند) را خوانده و یک مقدار ۰ یا ۱ (که **خروجی** دستور نامیده می‌شود) را تولید می‌کند. خروجی دستور i در خانه‌ی $i \times W + 0$ ذخیره می‌شود. همچنین ورودی‌های دستور i تنها می‌تواند از بین خانه‌های مربوط به تصویر و یا خروجی دستورات قبلی باشد، یعنی خانه‌های $0 \leq i \leq H \times W - 1$ باشند.

چهار نوع دستور وجود دارد:

- NOT: دقیقاً یک ورودی دارد. خروجی آن ۱ است اگر ورودی ۰ باشد، در غیر این صورت خروجی آن ۰ است.
- AND: یک یا چند ورودی دارد. خروجی آن ۱ است اگر و تنها اگر همه‌ی ورودی‌های آن ۱ باشد.
- OR: یک یا چند ورودی دارد. خروجی آن ۱ است اگر و تنها اگر حداقل یکی از ورودی‌های آن ۱ باشد.
- XOR: یک یا چند ورودی دارد. خروجی آن ۱ است اگر و تنها اگر تعداد فردی از ورودی‌های آن ۱ باشد.

خروجی آخرین دستور برنامه در صورتی که فاصله‌ی بین دو پیکسل سیاه دقیقاً K باشد، باید برابر با ۱ بوده و در غیر این صورت باید برابر با ۰ باشد.

جزئیات پیاده‌سازی

شما باید تابع زیر را پیاده‌سازی کنید:

```
void construct_network(int H, int W, int K)
```

- H, W : ابعاد تصویری که توسط دوربین روبات گرفته می‌شود
- K : یک عدد صحیح مثبت
- این تابع باید یک برنامه برای روبات تولید کند. برای هر تصویر که توسط دوربین روبات گرفته می‌شود، این برنامه باید مشخص کند که فاصله‌ی بین دو پیکسل سیاه در تصویر دقیقاً برابر با K است یا خیر.

این تابع باید یک یا چند تابع زیر را برای افزودن دستورات به برنامه‌ی روبات (که در ابتداء خالی است) فراخوانی کند:

```
int add_not(int N)
int add_and(int[] Ns)
int add_or(int[] Ns)
int add_xor(int[] Ns)
```

- افزودن یک دستور NOT، AND، OR یا XOR (به ترتیب).
- N (برای add_not): اندیس خانه‌ای که دستور NOT افزوده شده، ورودی اش را از آن می‌خواهد.
- Ns (برای add_and, add_or, add_xor): آرایه‌ای شامل اندیس خانه‌هایی که دستورات AND، OR یا XOR افزوده شده، ورودی‌هاشان را آن می‌خوانند.
- هر کدام از این چهار تابع، اندیس خانه‌ای که خروجی دستور را ذخیره می‌کند، برمی‌گرداند. فراخوانی‌های متوالی به این توابع، اعداد متوالی با شروع از $H \times W$ را برمی‌گردانند.

برنامه‌ی روبات می‌تواند از حداقل 10 000 دستور تشکیل شود. این دستورات می‌توانند حداقل در مجموع 1 000 000 مقدار را بخوانند. به عبارت دیگر، مجموع طول آرایه‌های Ns در تمام فراخوانی‌های add_and، add_not، add_xor و add_or به علاوه‌ی تعداد فراخوانی‌های add_or از 1 000 000 باشد.

پس از افزودن آخرین دستور، تابع construct_network باید (با برگرداندن مقدار تهی) به پایان برسد. پس از آن برنامه‌ی روبات بر روی تعدادی تصویر ارزیابی می‌شود. پاسخ شما روی یک داده‌ی آزمون پذیرفته می‌شود اگر برای هر تصویر، خروجی آخرین دستور 1 باشد اگر و تنها اگر فاصله‌ی بین دو پیکسل سیاه در تصویر دقیقاً برابر با K باشد.

ارزیابی پاسخ شما ممکن است به یکی از پیام‌های خطای زیر منجر شود:

- Instruction with no inputs: یک آرایه‌ی خالی به عنوان ورودی add_and، add_or یا add_and، add_or داده شده است.
- Invalid index: یک اندیس اشتباه (که ممکن است منفی باشد) به عنوان ورودی add_and، add_or، add_and، add_or داده شده است.
- Too many instructions: تابع شما قصد داشته است بیش از 10 000 دستور اضافه کند.
- Too many inputs: دستورات در مجموع بیش از 1 000 000 مقدار ورودی را می‌خوانند.

مثال

فرض کنید 2×3 دو تصویر ممکن وجود دارد که فاصله‌ی پیکسل‌های سیاه ۳ باشد.

0	1	2
3	4	5

0	1	2
3	4	5

- حالت ۱: پیکسل‌های سیاه ۰ و ۵ باشند
- حالت ۲: پیکسل‌های سیاه ۲ و ۳ باشند

یک پاسخ ممکن، ساختن برنامه‌ی روبات با فراخوانی‌ها زیر است:

1. `add([0, 5])`, که یک دستور با خروجی 1 ایجاد می‌کند اگر و تنها اگر حالت اول برقرار باشد.
خروجی این دستور در خانه‌ی 6 ذخیره می‌شود.
2. `add_and([2, 3])`, که یک دستور با خروجی 1 ایجاد می‌کند اگر و تنها اگر حالت دوم برقرار باشد.
خروجی این دستور در خانه‌ی 7 ذخیره می‌شود.
3. `add_or([6, 7])`, که یک دستور با خروجی 1 ایجاد می‌کند اگر و تنها اگر یکی از دو حالت بالا برقرار باشد.

محدودیت‌ها

$$\begin{aligned} 1 \leq H \leq 200 \\ 1 \leq W \leq 200 \\ 2 \leq H \times W \\ 1 \leq K \leq H + W - 2 \end{aligned}$$

زیرمسئله‌ها

- .1 $\max(H, W) \leq 3$ (۱۰ امتیاز)
- .2 $\max(H, W) \leq 10$ (۱۱ امتیاز)
- .3 $\max(H, W) \leq 30$ (۱۱ امتیاز)
- .4 $\max(H, W) \leq 100$ (۱۵ امتیاز)
- .5 $\min(H, W) = 1$ (۱۲ امتیاز)
- .6 (۸ امتیاز) پیکسل سطر ۰ و ستون ۰ در تصویر سیاه است.
- .7 (۱۴ امتیاز) $K = 1$
- .8 (۱۹ امتیاز) بدون محدودیت اضافی.

ارزیاب نمونه

ارزیاب نمونه ورودی را در قالب زیر می‌خواند:

- خط $H \ W \ K : 1$
- خط $r_1[i] \ c_1[i] \ r_2[i] \ c_2[i] :(i \geq 0) 2 + i$
- خط آخر: $-1 \ c_2[i]$

هر خط به جز خط اول و آخر، نشانده‌نده تصویری با دو پیکسل سیاه است. تصویری که در خط $i + 2$ توصیف شده است، تصویر i نامیده می‌شود. یک پیکسل سیاه در سطر $r_1[i]$ و ستون $c_1[i]$ و دیگری در سطر $r_2[i]$ و ستون $c_2[i]$ قرار دارد.

ارزیاب نمونه ابتدا `construct_network(H, W, K)` را فراخوانی می‌کند. اگر `construct_network(H, W, K)` از محدودیت‌های توصیف شده در صورت سؤال را نقض کند، ارزیاب نمونه یکی از پیام‌های خطایی که در پایان قسمت جزئیات پیاده‌سازی ذکر شده است را چاپ کرده و خارج می‌شود.

در غیر این صورت، ارزیاب نمونه دو خروجی زیر را تولید می‌کند.

نخست، ارزیاب نمونه خروجی برنامه‌ی روبات را در قالب زیر چاپ می‌کند:

- خط $i + 1 \leq 0$: خروجی آخرین دستور برنامه‌ی روبات برای تصویر i (۱ یا ۰).

دوم، ارزیاب نمونه یک فایل `log.txt` را درون شاخه‌ی جاری در قالب زیر می‌نویسد:

- خط $i + 1 \leq 0$: $m[i][0] \ m[i][1] \ \dots \ m[i][c - 1] :(0 \leq i)$

دنباله‌ی موجود در خط $i + 1$ مقادیر ذخیره شده در خانه‌های حافظه‌ی روبات را پس از اجرای برنامه‌ی روبات توصیف می‌کند، با فرض این که تصویر i به عنوان ورودی داده شده باشد. به طور خاص، $[j][i]$ برابر با مقدار خانه‌ی j حافظه است. توجه کنید که مقدار c (طول دنباله) برابر با $H \times W$ به علاوه‌ی تعداد دستورات برنامه‌ی روبات است.