

زمان مجاز برای هر مورد: ۱ ثانیه
حافظه‌ی مصرفی مجاز: ۲۰۰ مگابایت

تحقیقات اخیر بر روی سیاره‌ی «سکونیل» حاکی از وجود نوع خاصی از موجودات زنده موسوم به «ابونتو» در این سیاره است. با توجه به اطلاعات به دست آمده ابونتوها نیز از دو جنس نر و ماده هستند؛ نرها صورتی و ماده‌ها سبز هستند. نکته‌ی جالب توجه در مورد ابونتوها این است که آنها «همبند» نیستند؛ هر یک از ابونتوها از تعدادی مؤلفه‌ی همبند موسوم به «پاره» تشکیل شده که همیشه در کنار هم حرکت می‌کنند و اگر از هم دور شوند باعث مرگ آن ابونتو می‌شود. یک ابونتو هنگام تولد دقیقاً k پاره است که به مرور زمان بزرگتر شده و پاره‌های جدیدی پیدا می‌کند. ابونتوها هنگام ورزش صبحگاهی در دو ستون، کنار هم می‌ایستند و ورزش می‌کنند. این ستون‌ها از پاره‌های سبز (مربوط به ماده‌ها) و صورتی (مربوط به نرها) تشکیل شده‌است. هر کدام از ستون‌ها دقیقاً شامل n پاره است. پاره‌های یک ابونتو می‌توانند در هر دو ستون وجود داشته باشند. یک پاره از یک ابونتو که در i امین مکان یک ستون ایستاده با پاره‌ی $1 - i$ ام و $1 + i$ ام از همان ستون و همچنین پاره‌ی i ام از ستون دیگر همسایه به حساب می‌آید. با توجه به همسایگی گفته شده می‌دانیم پاره‌های یک ابونتوی در حال ورزش صبحگاهی تشکیل یک شکل همبند می‌دهند.

عکسی توسط یک تلسکوپ بسیار قوی از ابونتوها در حال ورزش صبحگاهی گرفته شده است. ولی در عکسی که به دست ما رسیده به دلیل مشکلات عکس برداری، رنگ تعدادی از پاره‌ها اشتباه افتاده است (سبز به جای صورتی یا صورتی به جای سبز). با توجه به اینکه می‌دانیم در عکس گرفته شده همه‌ی ابونتوها زنده هستند و همچنین می‌دانیم که هر ابونتو حداقل از k پاره تشکیل شده است، یک عکس می‌تواند نامعتبر باشد. می‌خواهیم شبیه‌ترین عکس (عکس با کمترین تعداد تغییر) به عکس فعلی را پیدا کنیم که معتبر باشد. دقت کنید که اگر دو ابونتوی صورتی (یا سبز) کنار هم ایستاده باشند، قادر به تشخیص پاره‌های آنها از هم نیستیم و حتی نمی‌توانیم تشخیص دهیم که این یک ابونتو است یا دو ابونتو.

مسئله

برنامه‌ای بنویسید که

- n و k و اطلاعات عکس را از ورودی استاندارد بخواند؛
- مشخص کند که عکس گرفته شده با شبیه‌ترین عکس معتبر چند تفاوت دارد؛ و
- تعداد تفاوت‌ها و شبیه‌ترین عکس معتبر (اگر چند جواب وجود دارد یکی از آنها را به دلخواه) را در خروجی استاندارد بنویسد.

می‌توانید فرض کنید که ورودی‌های داده‌شده به برنامه‌ی شما حتماً دارای جواب هستند.

ورودی

سطر نخست ورودی به ترتیب شامل n و k است ($1 \leq n \leq 1000$ و $1 \leq k \leq 100$). سطر دوم و سوم هر کدام شامل n عدد هستند که با یک فاصله از هم جدا شده‌اند و به ترتیب نشان دهنده‌ی رنگ پاره‌های ستون اول و دوم عکس هستند؛ در این سطرها صفر نشان دهنده‌ی رنگ صورتی و یک نشان دهنده‌ی رنگ سبز می‌باشد.

خروجی

در سطر اول خروجی تعداد تفاوت‌های عکس اولیه با شبیه‌ترین عکس معتبر به آن را بنویسید؛ این عدد را با m نشان می‌دهیم. در هر یک از m سطر بعدی دو عدد بنویسید که محل یکی از پاره‌هایی که باید تغییر رنگ دهد، را مشخص می‌کند؛ ابتدا ۱ یا ۲ که نشان دهید این پاره در کدام ستون است و سپس ۱ تا n که نشان دهید کدام پاره از این ستون باید تغییر کند. سمت چپ‌ترین پاره‌ای که در ورودی برای هر ستون آمده، با شماره‌ی ۱ نشان داده می‌شود.

ورودی و خروجی نمونه

Standard Input	Standard Output
5 3	3
0 1 0 1 0	1 1
1 0 1 0 1	1 5
	2 3

با انجام تغییراتی که در خروجی نمونه آمده، عکس به صورت زیر تغییر می‌یابد.

<u>1</u>	1	0	1	<u>1</u>
1	0	<u>0</u>	0	1

شیوه ارزیابی

اگر برنامه‌ی تان برای یک مورد از حدود خود تجاوز نکند، و پاسخ صحیح (حداقل تعداد اشتباهات عکس) بدهد، نیمی از نمره را می‌گیرد. اگر در ادامه محل پاره‌هایی که باید تغییر رنگ دهد را نیز به درستی نوشته باشد، بقیه‌ی نمره را کسب می‌کند؛ ولی اگر هرگونه اشتباهی، چه در ساختاردهی خروجی یا صحت پاسخ این بخش داشته باشد، حداکثر می‌تواند نیمی از نمره را دریافت کند.