- سؤال‌های **۱۲ تا ۲۵** در دسته‌های چندسؤالی آمده‌اند و توضیح هر دسته پیش از آن آمده است.
- امتیاز همه‌ی سؤال‌ها یکسان است.
- جواب درست به هر سؤال چهار نمره‌ی مثبت و جواب نادرست یک نمره‌ی منفی دارد.
- ترتیب گزینه‌ها در هر سؤال به شکل تصادفی است.

۱ رستم ۱۳۹۴ سکه با شماره‌های ۱ تا ۱۳۹۴ بر روی میز قرار داده است. تعدادی از این سکه‌ها عادی (یک رو شیر و یک رو خط) و بقیه‌ی سکه‌ها هر دو رو شیر هستند (این تعداد می‌تواند صفر هم باشد). سهراب می‌خواهد تعداد سکه‌های هر دو رو شیر را پیدا کند ولی چشمانش بسته است. او تنها می‌تواند در هر حرکت تعدادی از سکه‌ها را انتخاب کرده و از رستم بخواهد آنها را پشت و رو کند. پس از آن رستم تعداد سکه‌های روی میز که به سمت شیر هستند را به سهراب می‌گوید.
سهراب می‌داند در ابتدای کار دقیقا ۱۰۰ سکه به سمت شیر هستند، حداقل چند حرکت لازم است تا سهراب تعداد سکه‌های دو رو شیر را بیابد؟

۱) ۱۳۹۳ ۲) ۱ ۳) ۱۳۹۴ ۴) ۱۱ ۵) ۱۰

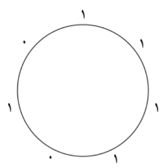
۲ می‌خواهیم در خانه‌های جدول زیر، اعداد ۱ تا ۹ را قرار دهیم، به صورتی که مجموع اعداد هر سطر، هر ستون و هر قطر، برابر باشد. جای دو تا از اعداد (۱ و ۵) نیز مشخص شده است.

	۵	۱

برای یک خط مانند L در صفحه، $f(L)$ برابر مجموع اعداد خانه‌هایی از جدول است که با آن خط، تقاطع دارند (یک خانه از جدول با خط L تقاطع دارد، اگر حداقل ۲ نقطه‌ی مشترک با آن خط داشته باشد). بیشینه‌ی ممکن $f(L)$، در میان تمام جدول‌ها و خط‌های ممکن چند است؟ (هر خانه از جدول یک مربع به طول واحد است)

۱) ۲۵ ۲) ۳۲ ۳) ۳۱ ۴) ۲۷ ۵) ۳۰

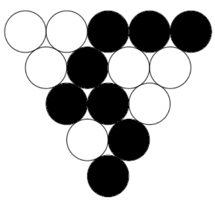
۳ می‌خواهیم ۷ رقم ۰ و ۱ را دور دایره بچینیم. می‌گوییم رشته‌ی S در این چینش آمده است، اگر چند رقم متوالی در دایره وجود داشته باشند که با کنار هم قرار دادن‌شان به ترتیب ساعت‌گرد، رشته‌ی S تشکیل شود. تعداد دفعات وجود S در چینش را $f(S)$ می‌نامیم.



برای مثال، در چینش روبرو، $f(۱۱۰) = ۱$ و $f(۱۱) = ۳$ و $f(۰۱۱۱۰) = ۰$ است.
یک چینش اعداد دور دایره را در نظر بگیرید. به ازای هر رشته‌ی دودویی S که حداکثر ۳ رقم دارد، $۲^{f(S)}$ را محاسبه می‌کنیم و این مقادیر را با هم جمع می‌کنیم (به عنوان مثال در شکل مقابل این عدد برابر ۷۰ می‌شود). عدد نهایی حداقل چند است؟ (برای رشته‌هایی که در چینش وجود ندارند $f(S) = ۰$ است.)

۱) ۵۵ ۲) ۵۱ ۳) ۵۶ ۴) ۶۳ ۵) ۵۳

۴ ۱۵ دایره همانند شکل روبرو داریم. هر دایره می‌تواند سفید یا سیاه باشد. رنگ دایره‌ها به صورت زیر مشخص می‌گردد:



- دایره‌های سطر بالا به صورت مستقل می‌توانند سفید یا سیاه باشند.
- بقیه‌ی دایره‌ها (همه به جز سطر بالا) به رنگ سیاه هستند، اگر و تنها اگر دو دایره‌ی مجاور سطر بالای آن ناهمرنگ باشند.

در بین تمامی حالات ممکن، حداکثر چند دایره‌ی سیاه می‌توانیم داشته باشیم؟

۱) ۱۰ ۲) ۱۱ ۳) ۹ ۴) ۱۲ ۵) ۱۳

۵ در مسئله‌ی قبل، فرض کنید تمامی حالات ممکن را روی تخته کشیده‌ایم. در مجموع چند دایره‌ی سیاه خواهیم داشت؟

۱) ۲۴۰   ۲) ۲۱۶   ۳) ۲۵۶   ۴) ۲۰۸   ۵) ۲۲۴

۶ یک جایگشت

$$\pi = \langle \pi_1, \pi_2, \dots, \pi_9 \rangle$$

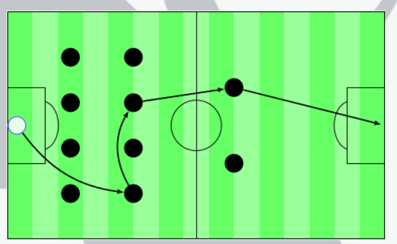
از اعداد ۱, ۲, ..., ۹ را در نظر بگیرید. عدد جایگشت π برابر تعداد اعضایی از جایگشت مانند π_i است که زوجیت i و π_i برابر باشد. برای مثال عدد جایگشت $\langle 5, 6, 3, 4, 2, 1, 7, 9, 8 \rangle$ برابر ۵ است. با در نظر گرفتن تمام جایگشت‌های ممکن، به طور میانگین عدد یک جایگشت چند است؟

۱) $\frac{81}{19}$ ۲) $\frac{13}{3}$ ۳) $\frac{41}{9}$ ۴) ۵   ۵) $\frac{9}{2}$

۷ یک عدد را **وارونه** می‌گوییم، هر گاه به صورت $\frac{1}{n}$ باشد که n عددی طبیعی است. می‌خواهیم عدد ۱ را به صورت جمع k عدد وارونه‌ی متمایز بنویسیم. به ازای چند مقدار $2 \leqslant k \leqslant 6$ می‌توان این کار را انجام داد؟

۱) ۳   ۲) ۱   ۳) ۵   ۴) ۰   ۵) ۴

۸ تیم فوتبال سلطان، با سیستم $4 - 4 - 2$ بازی می‌کند؛ یعنی ۱ دروازه‌بان، ۴ مدافع و ۴ هافبک و ۲ مهاجم دارد. هر توپی که به یک بازیکن در این تیم می‌رسد، یا آن را با یک شوت، تبدیل به گل می‌کند یا پاس می‌دهد.



هیچ بازیکنی حق ندارد به بازیکنی پاس بدهد که قبلا توپ به او رسیده و یا در خطوط عقب‌تر بازی می‌کند؛ برای مثال یک هافبک نمی‌تواند به یک مدافع پاس بدهد، اما می‌تواند به یک هافبکی که توپ به آن نرسیده و یا یک مهاجم پاس بدهد.
فرض کنید توپ در ابتدا در اختیار دروازه‌بان است و تیم می‌خواهد یک گل بزند (همانند شکل زیر). به چند طریق می‌توان این کار را انجام داد؟ (حتی دروازه‌بان هم می‌تواند با یک ضربه‌ی مستقیم گل بزند.)

۱) ۲۳۰۴   ۲) ۱۱۵۲   ۳) ۵۰۴۳   ۴) ۲۸۶۲۵   ۵) ۲۱۱۲۵

۹ سه توپ سیاه و سه توپ سفید داریم که به شکل زیر، در هفت جعبه جای گرفته‌اند:

فاصله‌ی دو جعبه تعداد جعبه‌های بین آن دو است. برای مثال فاصله‌ی دو جعبه‌ی مجاور صفر است. در هر حرکت می‌توان یک توپ که فاصله‌ی جعبه‌اش با یک جعبه‌ی خالی، حداکثر یک است را به خانه‌ی خالی انتقال داد. می‌خواهیم به حالتی برسیم که سه توپ سفید در سه جعبه‌ی سمت چپ و سه توپ سیاه در سه جعبه‌ی سمت راست باشند. حداقل چند حرکت برای این کار لازم است؟

۱) ۱۴ ۲) ۱۳ ۳) ۱۵ ۴) ۱۷ ۵) ۱۶

۱۰ جایگشت $a_1, a_2, \cdots, a_6$ از اعداد ۱ تا ۶ را در نظر بگیرید. در ابتدا یک عدد را به دل‌خواه انتخاب می‌کنیم و سپس در هر مرحله اگر عدد a_i انتخاب شده بود در مرحله‌ی بعد به ازای $a_i \neq 6$ عدد a_{a_i+1} و برای $a_i = 6$ عدد a_1 انتخاب می‌شود. به ازای چند جایگشت مختلف می‌توان عدد اول را به گونه‌ای انتخاب کرد که بعد از تعدادی مرحله، همه‌ی اعداد جایگشت حداقل یک‌بار انتخاب شده باشند؟

۱) ۳۴۵ ۲) ۰ ۳) ۲۴۰ ۴) ۷۲۰ ۵) ۱۲۰

۱۱ مجموعه‌ی $S = \{۱, ۲, ..., ۷\}$ داده شده است. دو تابع داریم:
$f(A)$ که مکمل زیرمجموعه‌ی A و $g(A, B)$ که اشتراک A و B را می‌دهد.
یک کیسه داریم که همه‌ی زیرمجموعه‌های S را درون آن ریخته‌ایم. می‌خواهیم تعدادی زیرمجموعه از کیسه بیرون آوریم تا با استفاده از آن‌ها و توابع بتوان تمام زیرمجموعه‌های S را تولید کرد (برای تولید زیرمجموعه‌ها می‌توان از هر زیرمجموعه به تعداد دل‌خواه استفاده کرد).
فرض کنید $\{۱, ۲, ۵, ۶\}$، $\{۲, ۵, ۳\}$ و $\{۵, ۶\}$ را از کیسه بیرون آورده‌ایم. حداقل چند زیرمجموعه‌ی دیگر از کیسه بیرون بکشیم تا مطمئن باشیم با آن‌ها می‌توان مسئله را حل کرد؟

۱) ۱ ۲) ۶۲ ۳) ۶۷ ۴) ۶۴ ۵) ۳

یک کلمه درون یک پرونده‌ی متنی داده شده است و می‌خواهیم با کم‌ترین تعداد اعمال کپی (Copy) و درج (Paste) تعداد مشخصی نسخه از آن ایجاد کنیم.
در هر مرحله می‌توانیم تعداد دل‌خواهی از کلمات نوشته‌شده درون پرونده را در حافظه کپی کنیم و یا کلمات درون حافظه را در پرونده درج کنیم. (مثل کپی و پیست در ویرایشگرها، می‌توان یک بار کپی کرد و سپس چند بار درج کرد.)
هر کپی ۱ واحد و هر درج نیز ۱ واحد هزینه دارد.

با توجه به توضیحات بالا به ۲ سؤال زیر پاسخ دهید

۱۲ با حداقل چند واحد هزینه می‌توانیم دقیقا ۹۹ کلمه‌ی دیگر مشابه با کلمه‌ی اول ایجاد کنیم؟

۱) ۱۳ ۲) ۲۰ ۳) ۱۴ ۴) ۱۰ ۵) ۲۲

۱۳ با ۱۴ واحد هزینه حداکثر چند کلمه (با احتساب کلمه‌ی اولیه) می‌توان ایجاد کرد؟

۱) ۱۰۰ ۲) ۱۲۸ ۳) ۲۴۳ ۴) ۸۱ ۵) ۱۶۲

فرض کنید در یک کشور، اسکناس‌های a_1 تومانی، a_2 تومانی و ... داریم و بخواهیم مقدار n تومان را پرداخت کنیم (پرداخت یک‌طرفه است یعنی نمی‌توانیم مقداری را بپردازیم و بقیه‌ی پول را پس بگیریم). در صورتی که بتوان n تومان را پرداخت کرد، n را عددی **خوب** می‌گوییم. برای مثال اگر اسکناس‌های ۵۰, ۱۰۰, ۲۰۰, ۵۰۰ و ۱۰۰۰ تومانی داشته باشیم، ۲۹۰۰ عددی خوب است؛ اما ۲۹۵۳ عددی خوب نیست. هم‌چنین عددی مانند n را **عجیب** گوییم، اگر بتوان n تومان را پرداخت کرد؛ طوری که از هر نوع اسکناس حداکثر یک بار استفاده شود. در مثال قبل ۹۰۰ عجیب نیست.
اگر n یک عدد خوب باشد، کمینه‌ی تعداد اسکناس‌ها برای پرداختش را $f(n)$ می‌نامیم. فرض کنید یک نفر الگوریتم زیر را برای پرداخت انتخاب کند:

در هر مرحله بزرگ‌ترین اسکناسی که مقدار آن از n بیش‌تر نیست را انتخاب می‌کنیم. این مبلغ را پرداخت می‌کنیم و برای باقی پول همین روش را ادامه می‌دهیم تا پرداخت به طور کامل انجام شود.
عدد n را **زیبا** گوییم، اگر تعداد اسکناس‌هایی که با الگوریتم بالا پرداخت می‌کنیم، برابر $f(n)$ شود. به یک کشور، **افسانه‌ای** گوییم، اگر تمام اعداد طبیعی خوب، زیبا نیز باشند.

با توجه به توضیحات بالا به ۲ سؤال زیر پاسخ دهید

۱۴ فرض کنید در یک کشور، اسکناس‌های $۱, ۳, ۳^۲, ۳^۳, \dots$ تومانی داشته باشیم. می‌خواهیم یک نوع اسکناس از بین گزینه‌های زیر به اسکناس‌های‌مان اضافه کنیم. با اضافه کردن کدام گزینه تعداد اعداد عجیب n که $۱ \leqslant n \leqslant ۲۴۹$ بیش‌تر از بقیه گزینه‌ها است؟

۱) ۶ ۲) ۴ ۳) ۵ ۴) ۷ ۵) ۲

۱۵ فرض کنید گزینه‌های زیر اسکناس‌های ۵ کشور مختلف باشند. کدام گزینه مربوط به یک کشور افسانه‌ای نیست؟

۱) $۱, ۲, ۴, ۸, \dots$
۲) $۱!, ۲!, ۳!, \dots$
۳) $۱, ۲, ۳, ۵, ۹, \dots$ (۱ و $(۱ + ۲^n)$ها)
۴) $۱, ۴, ۹, ۱۶, \dots$
۵) گزینه‌های ۳ و ۴

گراف ساده‌ی n رأسی G با رئوس $۱, ۲, \dots, n$ را در نظر بگیرید. **ماتریس مسیریاب گراف**، یک ماتریس $n \times n$ است که درایه‌ی سطر iام و ستون jام آن، **تعداد مسیرهای بین رأس i و رأس j** است (مسیر دنباله‌ای از رئوس است که بین هر دو رأس متوالی یک یال وجود دارد و هر رأس حداکثر یک بار آمده است).
در صورتی که $i = j$ باشد، مقدار ۱ را در ماتریس قرار می‌دهیم.

با توجه به توضیحات بالا به ۳ سؤال زیر پاسخ دهید

۱۶ کدام یک از ماتریس‌های زیر، می‌تواند یک ماتریس مسیریاب باشد؟

ماتریس ۱:

۱	۱	۱	۲	۱
۱	۱	۱	۲	۱
۱	۱	۱	۱	۱
۲	۲	۱	۱	۱
۱	۱	۱	۱	۱

ماتریس ۲:

۱	۳	۳	۳	۳
۳	۱	۳	۳	۳
۳	۳	۱	۳	۳
۳	۳	۳	۱	۳
۳	۳	۳	۳	۱

ماتریس ۳:

۱	۱	۲	۱
۱	۱	۲	۲
۲	۲	۱	۲
۱	۲	۲	۱

ماتریس ۴:

۱	۴	۳	۳
۴	۱	۳	۳
۳	۳	۱	۳
۳	۳	۳	۱

ماتریس ۵:

۱	۳	۳	۲
۳	۱	۲	۳
۳	۲	۱	۳
۲	۳	۳	۱

۱) ماتریس ۲ ۲) ماتریس ۳ ۳) ماتریس ۴ ۴) ماتریس ۱ ۵) ماتریس ۵

۱۷ می‌خواهیم با پرسیدن تعدادی از خانه‌های ماتریس مسیریاب یک گراف، تعداد مؤلفه‌های گراف را تشخیص دهیم. در هر گام می‌توان یکی از خانه‌های ماتریس را پرسید. در حداقل چند گام به طور تضمینی به هدف می‌رسیم؟

۱) n ۲) $n-۱$ ۳) $n-۲$ ۴) $\binom{n-۱}{۲}+۱$ ۵) $\binom{n}{۲}$

۱۸ با استفاده از ماتریس مسیریاب یک گراف، پاسخ چند تا از موارد زیر را همواره می‌توان فهمید؟ (رأس برشی، رأسی است که پس از حذف آن تعداد مؤلفه‌های همبندی گراف افزایش یابد.)

- آیا گراف رأس برشی دارد؟
- آیا رأس v برشی است؟
- بین دو رأس v و u یال وجود دارد یا نه؟
- آیا گراف حداقل ۲ (نه لزوما مجزا) دور دارد؟

۱) ۱ ۲) ۳ ۳) ۰ ۴) ۲ ۵) ۴

روال جام حذفی بدین صورت است که ۲^n تیم در n مرحله با هم مسابقه می‌دهند به نحوی که در هر مرحله هر تیم با یک تیم دیگر مسابقه می‌دهد (مثلا در مرحله‌ی اول $۲^{n-۱}$ مسابقه انجام می‌شود) و تیم‌هایی که شکست بخورند حذف می‌شوند. تیم‌های پیروز شده (نیمه‌ی دیگر تیم‌ها) به مرحله‌ی بعد می‌روند و دوباره به همین ترتیب مسابقه می‌دهند تا جایی که فقط یک تیم باقی بماند که تیم قهرمان نامیده می‌شود.
هر تیم عددی بین ۰ تا $۲^n - ۱$ دارد که قدرت آن تیم را نیز نشان می‌دهد (قدرت هیچ دو تیمی با هم برابر نیست). در مسابقه‌ی بین دو تیم، تیمی پیروز خواهد شد که قدرت بیشتری داشته باشد مگر در شرایطی که هر سوال مشخص می‌کند.

با توجه به توضیحات بالا به ۳ سؤال زیر پاسخ دهید

۱۹ این مسابقات با شرکت ۶۴ تیم برگزار می‌شود ($n = ۶$). در یک مسابقه اگر قدرت دو تیم را در مبنای دو بنویسیم و همه‌ی رقم‌های آنها به جز یکی برابر باشند هر دو تیم ممکن است پیروز شوند، در غیر این صورت تیم قوی‌تر پیروز می‌شود. برای مثال اگر ۲ و ۱۰ با هم مسابقه بدهند، هر دو تیم می‌توانند پیروز شوند. اما اگر ۱۶ و ۷ با هم مسابقه بدهند، حتما ۱۶ پیروز خواهد شد. ضعیف‌ترین تیمی که ممکن است قهرمان شود چه تیمی است؟

۱) ۳۱ ۲) ۰ ۳) ۱۵ ۴) ۱ ۵) ۹

۲۰ در یک جام حذفی ۳۲ تیم حضور دارند ($n = ۵$) و چهار تیم ۳۱، ۲۳، ۱۴ و ۵ آمادگی کافی ندارند و ممکن است در یک مسابقه به طور اتفاقی شکست بخورند. با این شرایط ضعیف‌ترین تیمی که ممکن است قهرمان شود چه تیمی است؟

۱) ۱۵ ۲) ۱۶ ۳) ۱ ۴) ۴ ۵) ۰

۲۱ در یک جام حذفی ۱۶ تیم حضور دارند ($n = ۴$) و هر تیم ممکن است به طور اتفاقی در یک مسابقه پیروز شود. ضعیف‌ترین تیمی که ممکن است قهرمان شود چه تیمی است؟

۱) ۳ ۲) ۶ ۳) ۵ ۴) ۴ ۵) ۷

دنباله‌ای اکیدا صعودی مانند $a_۱, a_۲, \ldots, a_n$ از اعداد داریم. ما هیچ اطلاعاتی درباره‌ی اعداد نداریم و فقط می‌دانیم اکیدا صعودی هستند. عدد x در این دنباله موجود است اما نمی‌دانیم کجای دنباله است و می‌خواهیم مکان عدد

x در دنباله را بیابیم. در هر مرحله می‌توانیم یکی از a_iها را انتخاب کنیم؛ سپس به ما نتیجه‌ی مقایسه‌ی x با a_i گفته می‌شود؛ یعنی یکی از عبارات زیر گزارش داده می‌شود:

$$x < a_i, \quad x = a_i, \quad x > a_i$$

هزینه‌ی مقایسه‌ی عدد a_i با x، برابر w_i است. w_i داده شده است.
می‌خواهیم الگوریتمی ارائه دهیم که مکان عدد x در دنباله را بیابد. کمینه‌ی هزینه‌ای که بتوان به طور تضمینی این کار را انجام داد

$$f(w_1, w_2, ..., w_n)$$

می‌نامیم. برای مثال می‌توان نشان داد اگرتمام w_iها برابر ۱ باشند، این مقدار برابر $\lfloor \lg(n) \rfloor$ خواهد شد (منظور از $\lg(n)$، لگاریتم n در مبنای ۲ است).

با توجه به توضیحات بالا به ۴ سؤال زیر پاسخ دهید

۲۲ مقدار

$$f(\underbrace{2, 3, ..., 10, 1, 2, 3, ..., 10, ..., 1, 2, 3, ..., 10}_{\text{۳۱۹ عدد}})$$

چند است؟

۱) ۲۷   ۲) ۲۶   ۳) ۲۰   ۴) ۱۹   ۵) ۱۶

۲۳ مقدار

$$f(\underbrace{1, 1, ..., 1}_{\text{۵۱۱ عدد}}, \underbrace{2, 2, ..., 2}_{\text{۵۱۱ عدد}})$$

چند است؟

۱) ۱۱   ۲) ۱۹   ۳) ۲۷   ۴) ۱۲   ۵) ۱۸

۲۴ فرض کنید $n \geqslant 4$ باشد. مقدار

$$f(n, n^2, n^3, ..., n^n)$$

چند است؟

۱) $n^{n-3} + n^{n-1}$
۲) $\left\lfloor n^{\frac{n}{2}} + n^{\frac{3n}{4}} + n^{\frac{7n}{8}} + ... \right\rfloor$
۳) $\sum_{1 \leqslant 2k+1 \leqslant n} n^{2k+1}$
۴) $\lfloor \lg(1 + n + n^2 + ... + n^n) \rfloor$
۵) $n + n^2 + ... + n^{n-1}$

۲۵ فرض کنید $n \geqslant 3$ و تمام w_iها متمایز هستند. چند تا از گزاره‌های زیر همواره درست هستند؟

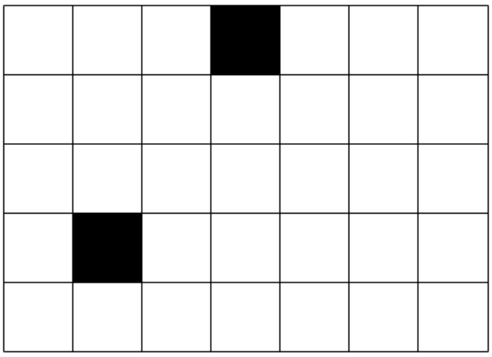
- هیچ الگوریتم بهینه‌ای در مرحله‌ی اول w_i بیشینه را انتخاب نمی‌کند.

- الگوریتم بهینه‌ای وجود دارد که در مرحله‌ی اول، w_iای را انتخاب می‌کند که $|\sum_{j<i} w_j - \sum_{j>i} w_j|$ کم‌ترین مقدار ممکن را داشته باشد.
- جواب بهینه‌ای وجود دارد که در هیچ مرحله‌ای، a_1 را انتخاب نکند.
- جواب بهینه‌ای وجود دارد که در مرحله‌ی اول، w_i کمینه را انتخاب کند.

۱) ۱   ۲) ۴   ۳) ۲   ۴) ۰   ۵) ۳

فرهاد و علی‌رضا در منهتن .... **۳۰ امتیاز**

فرهاد و علی‌رضایک جدول $m \times n$ دارند $(m, n > 1)$ و روی آن بازی می‌کنند. خانه‌ی واقع در سطر i-ام و ستون j-ام جدول را با (i, j) نشان می‌دهیم. **فاصله‌ی** دو خانه‌ی (r_1, c_1) و (r_2, c_2) را برابر $|r_1 - r_2| + |c_1 - c_2|$ تعریف می‌کنیم. برای مثال، در جدول زیر، فاصله‌ی دو خانه‌ی مشخص شده برابر ۵ است:



فرهاد k خانه‌ی $a_1, a_2, \ldots, a_k$ از جدول را انتخاب می‌کند و به علی‌رضا می‌گوید؛ سپس علی‌رضا یک خانه از جدول مانند X را انتخاب می‌کند و پس از این انتخاب، اعداد $b_1, b_2, \ldots, b_k$ را به فرهاد می‌گوید که b_i، فاصله‌ی خانه‌ی X تا خانه‌ی a_i است. در واقع علی‌رضا به ازای هر خانه‌ی انتخابی فرهاد، فاصله‌ی X را تا آن خانه به فرهاد می‌گوید. حال باید فرهاد با توجه به این k عدد، خانه‌ی مورد نظر علی‌رضا (X) را پیدا کند. فرهاد در صورتی می‌برد که خانه‌ی مورد نظر علی‌رضا را بفهمد. فرض کنید هر دو نفر به بهترین نحو ممکن بازی می‌کنند.

اگر k کم‌ترین عددی باشد که فرهاد، روشی برای انتخاب k خانه داشته باشد که ببرد، تعداد روش‌های انتخاب این k خانه که فرهاد با انتخاب آن‌ها، به هدفش می‌رسد را حساب کنید. توجه کنید تنها با یافتن k، می‌توانید ۱۵ نمره بگیرید.

زبان اعصاب .... ۵۰ امتیاز

فرهاد هم‌واره دوست داشت که توابع را به صورت ساده بیان کند. به همین دلیل، امروز به این نتیجه رسید که اکثر توابع را می‌توان با تعدادی تابع اولیه و عمل‌گر ساده پیاده‌سازی کرد. از شما می‌خواهیم که به فرهاد در پیاده‌سازی برخی از این توابع کمک کنید. هم‌چنین فرهاد تنها به توابعی علاقه دارد که ورودی و خروجی آن‌ها، اعدادی صحیح و نامنفی هستند. علی‌رضا به فرهاد توابع ساده‌ی اولیه‌ی زیر را پیشنهاد داده است:

۱. **تابع پوچ**: این تابع تنها یک ورودی می‌گیرد و در خروجی، عدد 0 را تحویل می‌دهد. این تابع را با z نشان می‌دهیم. برای مثال $z(10) = 0$.

۲. **تابع افزون‌گر**: این تابع تنها یک ورودی می‌گیرد و اگر عدد n در ورودی به آن داده شود، عدد $n+1$ را به عنوان خروجی تحویل می‌دهد. این تابع را با inc نشان می‌دهیم. برای مثال $inc(5) = 6$. فرهاد برای سادگی نمادی نیز تعریف کرده است. او به جای $x+1$ یا همان $inc(x)$ از نماد x' استفاده می‌کند.

۳. **توابع بازتاب**: این توابع به صورت $P_i^n(a_1, a_2, \dots, a_n)$ هستند که n عدد از ورودی می‌گیرند و i-امین عدد را تحویل می‌دهند. برای مثال تابع P_3^{10} تابعی است که با گرفتن 10 ورودی، هم‌واره سومین ورودی را برمی‌گرداند. به عنوان مثالی دیگر $P_2^3(0, 10, 8) = 10$ است.

با توابع بالا به تنهایی کار خاصی نمی‌توان کرد. به همین دلیل علی‌رضا عمل‌گرهای زیر را نیز به فرهاد پیشنهاد داده است. فایده‌ی این عمل‌گرها این است که با گرفتن چند تابع می‌توان توابع جدید ساخت.

۱. **عمل‌گر ترکیب**: این عمل‌گر یک تابع اصلی f می‌گیرد. فرض کنید f، m ورودی بگیرد. سپس این عمل‌گر توابع $g_1, g_2, \dots, g_m$ را نیز از ورودی تحویل می‌گیرد که g_i-ها ورودی‌های یک‌سانی می‌گیرند (مثلن $x_1, x_2, \dots, x_n$). به این ترتیب تابع جدید h ساخته می‌شود که $h(x_1, x_2, \dots, x_n)$،

$$f\big(g_1(x_1, x_2, \dots, x_n), g_2(x_1, x_2, \dots, x_n), \dots, g_m(x_1, x_2, \dots, x_n)\big)$$

را برمی‌گرداند. این عمل‌گر را با

$$CN[f, g_1, g_2, \dots, g_m]$$

نشان می‌دهیم.

۲. **عمل‌گر بازگشت**: این عمل‌گر، دو تابع f, g را می‌گیرد که f تابعی با یک ورودی و g تابعی با سه ورودی است. سپس این عمل‌گر، تابع بازگشتی h (با دو ورودی) را به صورت زیر می‌سازد:

$$\begin{cases} h(x, 0) = f(x) \\ h(x, y') = g\big(x, y, h(x, y)\big) \end{cases}$$

یادآوری می‌کنیم منظور از y' همان $y+1$ یا $inc(y)$ است. در واقع این عمل‌گر برای محاسبه‌ی $h(x,y')$، مقدار $h(x,y)$ را به صورت بازگشتی محاسبه می‌کند و سپس حاصل $g(x,y,h(x,y))$ را برمی‌گرداند. این عمل‌گر را با $\mathrm{PR}[f,g]$ نشان می‌دهیم.

فرهاد که حسابی گیج شده بود، از علی‌رضا خواست تا چند مثال برای او بزند. علی‌رضا دو مثال زیر را برای بهتر فهمیدن فرهاد ارائه کرد:

۱. فرض کنید می‌خواهیم تابع $const_1$ را بسازیم. این تابع باید به ازای هر ورودی x، هم‌واره عدد 1 را به عنوان خروجی تحویل دهد. قبل از پیاده‌سازی این تابع، به هدف پیاده‌سازی این تابع توجه کنید. توابع و عمل‌گرهای تعریف شده، بسیار ساده و مقدماتی هستند و شما حتی دسترسی مستقیم به یک عدد صحیح ندارید و حتی نمی‌توانید مستقیمن یک عدد صحیح به عنوان ورودی یک تابع بدهید؛ به همین دلیل برای ساختن عدد 1، این تابع را می‌سازیم. پس از پیاده‌سازی این تابع، دسترسی به عدد 1 خواهیم داشت. علی‌رضا روش زیر را برای پیاده‌سازی این تابع، پیشنهاد کرد:

$$const_1 = CN[inc, z]$$

این تابع در واقع با گرفتن عدد x، ابتدا آن را به تابع z و حاصل یا همان صفر را به تابع inc می‌دهد و در انتها خروجی ۱ برگردانده می‌شود.

فرض کنید $const_i$ تابعی باشد که با گرفتن هر عدد x، عدد ثابت i را برگرداند. علی‌رضا به این نکته توجه کرد که به ازای هر c ثابت، با داشتن تابع $const_{c-1}$، تابع $const_c$ را می‌توان به صورت زیر، پیاده‌سازی کرد:

$$const_c = CN[inc, const_{c-1}]$$

سپس با استقرا نتیجه گرفت به ازای هر c ثابت، تابع $const_c$ را می‌توان پیاده‌سازی کرد.

۲. فرض کنید می‌خواهیم تابع جمع (sum) را بسازیم. این تابع باید با گرفتن دو ورودی x,y، جمع آن‌ها $(x+y)$ را تحویل دهد. علی‌رضا برای پیاده‌سازی این تابع به صورت زیر استدلال کرد:

«در صورتی که $y=0$ باشد، آن‌گاه حاصل $x+y$ برابر x می‌شود؛ در غیر این صورت، حاصل $x+y$ برابر $sum(x,(y-1))+1$ است. پس می‌توان به صورت بازگشتی، این تابع را پیاده‌سازی کرد و کافی است دو تابع f,g را برای عمل‌گر بازگشت، تعریف کنیم:

- از آن‌جایی که $sum(x,0)=x$، پس باید تابع f را طوری تعریف کنیم که $f(x)=x$ شود. تابع بازتاب P_1^1 با گرفتن یک ورودی، همان را برمی‌گرداند؛ پس اگر قرار دهیم $f(x)=P_1^1(x)$، تابع مطلوب f را ساخته‌ایم. پس

$$f = P_1^1$$

- داریم $sum(x,y')=sum(x,y)+1$ و باید $sum(x,y')=g(x,y,sum(x,y))$ شود. با استفاده از تابع بازتاب P_3^3 روی ورودی‌های تابع g، می‌توانیم ورودی سوم آن یا همان $sum(x,y)$ را به دست آوریم. سپس اگر حاصل را به تابع inc بدهیم، مقدار مورد نظر یا همان $sum(x,y')$ ساخته می‌شود؛ پس اگر قرار دهیم

$$g(x,y,sum(x,y)) = inc\left(P_3^3(x,y,sum(x,y))\right)$$

تابع مطلوب g را ساخته‌ایم. پس $g = CN[inc, P_3^3]$.

پس توابع f,g را برای پیاده‌سازی توسط عمل‌گر بازگشت، ساختیم.»

علی‌رضا پس از استدلال بالا، پیاده‌سازی زیر را ارائه داد:

$$sum = PR\left[P_1^1, CN[inc, P_3^3]\right]$$

حال برای پیاده‌سازی توابع زیر، به فرهاد کمک کنید. توجه کنید فقط باید از توابع و عمل‌گرهای گفته شده، استفاده کنید. برای ساده نوشتن، می‌توانید چند تابع کمکی تعریف کرده و پیاده‌سازی کنید و در پیاده‌سازی تابع خواسته شده، از آن‌ها استفاده کنید. در هر قسمت توضیحی کوتاه (در حد چند جمله) نیز برای پیاده‌سازی خود بدهید.

الف) تابع ضرب (mul) را پیاده‌سازی کنید. این تابع باید دو عدد x, y به عنوان ورودی بگیرد و $x \times y$ را به عنوان خروجی تحویل دهد. در واقع باید $mul(x, y) = x \times y$ شود. (۱۰ امتیاز)

ب) تابع $poly$ را پیاده‌سازی کنید. این تابع باید یک عدد x به عنوان ورودی بگیرد و $x^2 + x + 2$ را به عنوان خروجی تحویل دهد. در واقع باید $poly(x) = x^2 + x + 2$ شود. (۱۰ امتیاز)

پ) تابع فاکتوریل ($fact$) را پیاده‌سازی کنید. این تابع باید عدد n را به عنوان ورودی بگیرد و $n!$ را به عنوان خروجی تحویل دهد. در واقع باید $fact(n) = n!$ شود. (۱۰ امتیاز)

ت) تابع مینیمم (min) را پیاده‌سازی کنید. این تابع باید عدد x, y را به عنوان ورودی بگیرد و عدد کوچک‌تر را از میان دو عدد x, y تحویل دهد. برای مثال اگر 3,4 به عنوان ورودی به این تابع داده شوند، باید عدد 3 و اگر 7,7 داده شوند، باید عدد 7 به عنوان خروجی داده شود. (۲۰ امتیاز)

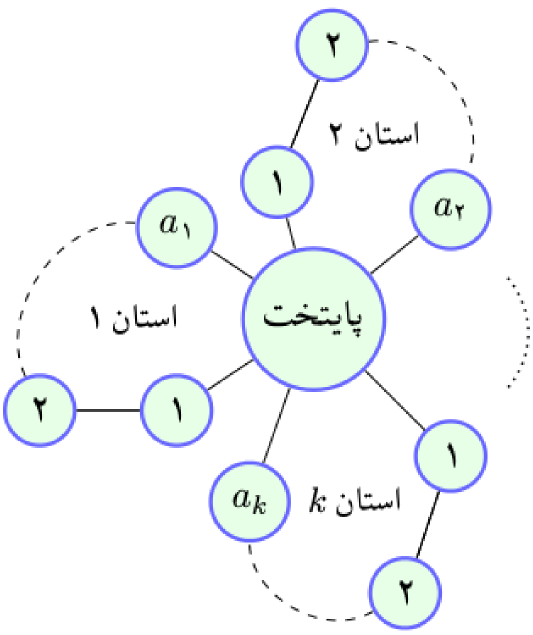
توپ‌های بهروز .... ۳۵ امتیاز

بهروز b جعبه و n نوع توپ دارد ($b \geq n$). در هر جعبه، تعدادی توپ وجود دارد. توجه کنید یک نوع توپ می‌تواند در چند جعبه وجود داشته باشد. می‌دانیم هر n جعبه‌ای را در نظر بگیریم، می‌توان از هر کدام، یک توپ انتخاب کرد؛ طوری که هیچ دو توپی از n توپ انتخاب شده، هم‌نوع نباشند. فرض کنید مجموع تعداد توپ‌های جعبه‌ها s باشد. کمینه‌ی ممکن s را بیابید (در واقع شما باید یک s پیدا کنید که حالتی با s توپ داشته باشیم؛ ولی هیچ حالتی با $s-1$ توپ وجود نداشته باشد).

بمب‌گذاری واس‌اوناس! .... ۶۵ امتیاز

در یک دنیا، هر کشور تعدادی شهر دارد و بین هر دو شهر، یا جاده‌ی مستقیم دوطرفه وجود دارد یا وجود ندارد. دو شهر را **مجاور** می‌گوییم، اگر با جاده‌ی مستقیم به هم وصل باشند. می‌دانیم از هر شهر می‌توان با طی کردن تعدادی جاده، به هر شهر دیگر رسید. **فاصله‌ی** بین دو شهر را کم‌ترین تعداد جاده‌هایی در نظر می‌گیریم که برای رفتن از یکی از این دو شهر به دیگری باید طی کرد.

الف) علی‌رضا و فرهاد، در کشور **واس‌ماس** زندگی می‌کنند. این کشور از یک پایتخت و k استان با شماره‌های $1, 2, \dots, k$ تشکیل شده است. استان i، a_i شهر با شماره‌های $1, 2, \dots, a_i$ دارد. در هر استان i، شهرهای ۱ و ۲ به هم، شهرهای ۲ و ۳ به هم، ... و شهرهای $a_i - 1$ و a_i به هم جاده دارند. هم‌چنین در هر استان i، شهرهای ۱ و a_i به پایتخت جاده دارند. در واقع جاده‌های این کشور مانند شکل زیر است:



یک تروریست در یکی از شهرهای این کشور، بمب‌گذاری کرده است. علی‌رضا و فرهاد که به تازگی پلیس شده‌اند، برای پیدا کردن شهر بمب‌گذاری شده، مأمور شده‌اند. آن‌ها یک دستگاه بمب‌یاب دارند. اگر در شهری مانند T این دستگاه را استفاده کنند، چنان‌چه شهر T بمب‌گذاری شده باشد، دستگاه به ما می‌گوید و اگر شهر T بمب‌گذاری نشده باشد، دستگاه در میان شهرهای مجاور T، شهری را نشان می‌دهد که کم‌ترین فاصله را با شهر بمب‌گذاری شده دارد (اگر چند شهر با این خاصیت وجود داشت، دستگاه به طور تصادفی یکی از آن‌ها را نشان می‌دهد). استفاده از این دستگاه، بسیار هزینه‌بر است؛ پس علی‌رضا و فرهاد می‌خواهند با کم‌ترین تعداد استفاده از دستگاه، شهر بمب‌گذاری شده را پیدا کنند. کم‌ترین تعداد دفعاتی که آن‌ها باید از دستگاه استفاده کنند تا بتوانند بمب را پیدا کنند، چقدر است؟ پاسخ را بر حسب اعداد $a_1, a_2, \dots, a_k$ بیان کنید. (۱۰ امتیاز)

ب) اتفاق ناگوار بمب‌گذاری، در کشور **ماس‌ماس** نیز رخ داد. پادشاه کشور ماس‌ماس تصمیم گرفته است از گروهی زبده برای خنثی کردن این بمب استفاده کند. پس از عمل‌کرد فوق‌العاده‌ی علی‌رضا و فرهاد در خنثی کردن بمب کشور واس‌ماس، پادشاه کشور ماس‌ماس تصمیم گرفت مأموریت را به این دو نفر و دست‌گاه عجیب‌شان بسپارد. پادشاه کشور ماس‌ماس به هر شهر، یک عدد نسبت داده است و آن عدد برابر با فاصله‌ی دورترین شهر کشور تا شهر مذکور است. علی‌رضا و فرهاد، اطلاعاتی در مورد تعداد شهرها و نحوه‌ی جاده‌کشی کشور ماس‌ماس ندارند. آن‌ها فقط می‌دانند کم‌ترین عدد نسبت داده شده به شهرها، r است. علی‌رضا و فرهاد ادعا می‌کنند پس از گرفتن نقشه‌ی جاده‌کشی کشور، خواهند توانست با حداکثر $f(r)$ بار استفاده از دست‌گاه، شهر بمب‌گذاری شده را پیدا کنند. پادشاه نیز پس از شنیدن این حرف، نقشه را به آن‌ها می‌دهد. با توجه به این که فرهاد و علی‌رضا هنگام بیان ادعا، تعداد شهرها و نقشه‌ی جاده‌کشی آن‌ها را نمی‌دانند، کمینه‌ی $f(r)$ را بیابید. (۲۵ امتیاز)

ج) عصبانیت بمب‌گذاران پس از خنثی شدن دو اقدام قبلی، دو چندان شد و آن‌ها این بار تصمیم گرفتند در کشور دوست و هم‌سایه (**باس‌ماس**) بمب‌گذاری کنند! این کشور n شهر دارد. ثابت کنید هر گونه شهرهای این کشور جاده‌کشی شده باشند، علی‌رضا و فرهاد می‌توانند با کم‌تر از $\lfloor lg(n) \rfloor + 1$ بار استفاده از دست‌گاه، شهر بمب‌گذاری شده را پیدا کنند.

توجه: منظور از $lg(n)$، لگاریتم n در مبنای ۲ است. برای مثال $lg(1024) = 10$ و ... $lg(20) = 4.32192$ است. هم‌چنین منظور از $\lfloor x \rfloor$، بزرگ‌ترین عدد صحیحی است که از x بزرگ‌تر نیست. برای مثال $\lfloor 2.3 \rfloor = 2$ و $\lfloor 5 \rfloor = 5$ است. (۳۰ امتیاز)

پاسخ تشریحی مرحله دوم بیست و پنجمین المپیاد کامپیوتر ایران

فرهاد و علی‌رضا در منهتن! .... ۳۰ امتیاز

□ ابتدا ثابت می‌کنیم کمینه‌ی k برابر ۲ است. ابتدا نشان می‌دهیم کمینه‌ی k نمی‌تواند ۱ باشد. اگر فرهاد تنها یک خانه را انتخاب کند و علی‌رضا پاسخ ۱ به او بدهد، از آن‌جایی که هر خانه دست کم ۲ خانه‌ی مجاور (ضلعی) دارد، پس فرهاد نمی‌تواند به طور یکتا خانه‌ی مورد نظر علی‌رضا را مشخص کند. حال نشان می‌دهیم فرهاد می‌تواند با انتخاب ۲ خانه، به هدفش برسد. فرض کنید فرهاد دو خانه‌ی $(1,1)$ و $(1,m)$ را انتخاب کند و علی‌رضا برای این دو خانه، به ترتیب پاسخ‌های p و q بدهد. اگر خانه‌ی مورد نظر علی‌رضا (r,c) باشد، باید $r+c=p+2$ و $c-r=q-(m-1)$ باشد. با حل دو معادله و دو مجهول بالا، r,c به طور یکتا دست می‌آید. ■

□ حال ثابت می‌کنیم تعداد روش‌های انتخاب ۲ خانه، طوری که فرهاد به هدفش برسد، ۴ است. اگر فرهاد دو خانه‌ی گوشه‌ای در طول یک ضلع را انتخاب کند، مانند روش بالا فرهاد می‌تواند به هدفش برسد. حال فرض کنید دو خانه به صورتی دیگر انتخاب شوند و این دو خانه، (r_1,c_1)، (r_2,c_2) باشند. ثابت می‌کنیم فرهاد نمی‌تواند به هدفش برسد. دو حالت داریم:

- فرض کنید این دو خانه، دو خانه در دو گوشه‌ی روبه‌روی جدول باشند. دو خانه‌ی مجاور (r_1,c_1) را در نظر بگیرید. اگر علی‌رضا یکی از این دو خانه را انتخاب کند، در هر صورت دنباله‌ی اعدادی که به فرهاد تحویل می‌دهد، یک‌سان است و فرهاد به هدفش نمی‌رسد.

- فرض کنید دست کم یکی از این دو خانه، در گوشه نباشند. بدون از دست دادن کلیت فرض کنید (r_1,c_1) در گوشه نباشد. پس حداقل ۳ خانه‌ی مجاور دارد. اگر فاصله‌ی دو خانه‌ی انتخابی فرهاد را d در نظر بگیریم، فاصله‌ی خانه‌ی (r_2,c_2) تا ۳ خانه‌ی مذکور تنها می‌تواند $d-1$ یا $d+1$ باشد. فاصله‌ی این ۳ خانه‌ی مذکور تا (r_1,c_1) نیز ۱ است. پس طبق اصل لانه کبوتر، حداقل دو تا از این خانه‌ها هستند که اگر علی‌رضا آن‌ها را انتخاب کند، دنباله‌ی اعداد یک‌سانی به فرهاد تحویل داده می‌شود و فرهاد به هدفش نمی‌رسد.

پس در هر حالت جز ۴ حالت گفته شده، فرهاد نمی‌تواند به هدفش برسد و پاسخ برابر ۴ است. ■

زبان اعصاب! .... ۵۰ امتیاز

الف) کاملن مانند مثال جمع در متن سوال عمل می‌کنیم. اگر $y=0$ باشد، $x\times y=0$ و در غیر این صورت $x\times y=x\times(y-1)+x$ است. پس اگر در عمل‌گر بازگشت، تابع f را برابر تابع z و تابع g را برابر $sum(x,mul(x,y-1))$ قرار دهیم، پیاده‌سازی انجام می‌شود. به دست آوردن $x,mul(x,y-1)$ جهت

دادن ورودی به تابع sum به راحتی توسط تابع P انجام می‌شود؛ زیرا هر دو مورد در ورودی‌های تابع g موجود هستند. پس داریم:

$$mul(x,y) = PR[z, CN[sum, P_1^3, P_3^3]]$$

ب) داریم $x^2 + x + 2 = x(x+1) + 2$. ابتدا $x \times (x+1)$ را محاسبه می‌کنیم. برای ساختن $x+1$ باید از تابع inc و برای ساختن $x(x+1)$ باید از تابع mul نوشته شده در قسمت قبل استفاده کنیم و آن‌ها را ترکیب کنیم. پس برای ساختن $x(x+1)$ کافی است تابع

$$tri(x) = CN[mul, P_1^1, inc]$$

را در نظر بگیریم. حال باید حاصل را با دو جمع کنیم. پس پاسخ برابر

$$f(x) = CN\Big[sum, tri, const_2\Big]$$

ج) ابتدا تابعی مانند $sudoFact(x,y)$ می‌نویسیم که $sudoFact(x,y) = y!$ شود. از عملگر بازگشت استفاده می‌کنیم. اگر $y = 0$ باشد، باید ۱ برگردانده شود؛ پس کافی است در عملگر بازگشت، f را برابر $const_1$ قرار دهیم که در متن سوال پیاده‌سازی شده است. اگر $y = 0$ نباشد، $sudoFact(x,y') = y' \times sudoFact(x,y)$ است. پس تابع g باید y' را در $sudoFact(x,y)$ ضرب کند. پیاده‌سازی زیر برای g این کار را انجام می‌دهد:

$$g(x, y, sudoFact(x,y)) = CN\Big[mul, P_3^3, CN[inc, P_2^3]\Big]$$

پس تابع $sudoFact$ را می‌توان به شکل زیر، پیاده‌سازی کرد:

$$sudoFact(x,y) = PR\Big[const_1, CN[mul, P_3^3, CN[inc, P_2^3]]\Big]$$

اکنون فقط کافی است تابع $sudoFact$ را به تابعی با یک ورودی تبدیل کنیم که عملگر $CN[sudoFact, P_1^1, P_1^1]$ این کار را برای ما انجام می‌دهد. پس تابع فاکتوریل را می‌توان به شکل زیر پیاده‌سازی کرد:

$$fact(n) = CN[sudoFact, P_1^1, P_1^1]$$

د) ابتدا تابع $dec(x)$ را تعریف می‌کنیم که با گرفتن x، عدد $x - 1$ را تحویل می‌دهد. البته در صورتی که $x = 0$ باشد، تابع باید عدد ۰ را تحویل دهد. این تابع به شکل زیر می‌تواند پیاده‌سازی شود:

$$dec(x) = CN\Big[PR[z, P_2^3], P_1^1, P_1^1\Big]$$

حال تابع $sub(x, y)$ را تعریف می‌کنیم که در آن اگر $x \leq y$ باشد، مقدار ۰ و در غیر این صورت مقدار $x - y$ باید برگردانده شود. با استفاده از تابع dec، تابع sub (تفریق) می‌تواند به شکل زیر پیاده‌سازی شود:

$$sub(x, y) = PR[P^{۱}_{۱}, CN[dec, P^{۳}_{۳}]]$$

حال تابع $sign(x)$ را تعریف می‌کنیم. اگر $x = ۰$ باشد، این تابع عدد ۰ و در غیر این صورت ($x > ۰$)، این تابع باید عدد ۱ را برگرداند. این تابع به شکل زیر می‌تواند پیاده‌سازی شود:

$$sign(x) = CN\left[sub, const_{۱}, CN[sub, const_{۱}, P^{۱}_{۱}]\right]$$

حال $\overline{sign}(x)$ را تعریف می‌کنیم که برعکس تابع $sign$ عمل می‌کند؛ یعنی اگر $x = ۰$ باشد، مقدار ۱ و در غیر این صورت مقدار ۰ را برمی‌گرداند. این تابع می‌تواند با استفاده از تابع $sign$ به صورت زیر نوشته شود:

$$\overline{sign}(x) = CN\left[sub, const_{۱}, CN[sign, P^{۱}_{۱}]\right]$$

حال تابع خواسته‌شده (min) را پیاده‌سازی می‌کنیم. با تعریف‌های بالا، به راحتی می‌توان بررسی کرد که

$$min(x, y) = sign(x - y) \times x + \overline{sign}(x - y) \times y$$

$sign(x - y) \times x$ را به صورت

$$case۱ = CN\Big[mul, CN[sign, CN[sub, P^{۲}_{۱}, P^{۲}_{۲}]], P^{۲}_{۱}\Big]$$

و $\overline{sign}(x - y) \times y$ را به صورت مشابه

$$case۲ = CN\Big[mul, CN[\,\overline{sign}, CN[sub, P^{۲}_{۱}, P^{۲}_{۲}]], P^{۲}_{۲}\Big]$$

پیاده‌سازی می‌کنیم. پس:

$$min(x, y) = CN[sum, case۱, case۲]$$

توپ‌های بهروز .... ۳۵ امتیاز

ثابت می‌کنیم پاسخ برابر $(b-n+1) \times n$ است.

□ ابتدا ثابت می‌کنیم در هر آرایش، حداقل این تعداد توپ داریم. هر نوع توپی که در نظر بگیرید، باید در حداقل $b-n+1$ جعبه آمده باشد. برهان خلف می‌زنیم. فرض کنید این طور نباشد. یعنی یک نوع توپ وجود دارد که در حداقل n جعبه نیامده است. آن n جعبه را در نظر بگیرید. نمی‌توان از آن‌ها توپ‌هایی انتخاب کرد که تمام انواع توپ‌ها انتخاب شوند و با فرض مسئله به تناقض می‌رسیم. پس $s \geq n \times (b-n+1)$ است. ■

□ حال ثابت می‌کنیم آرایشی با $(b-n+1) \times n$ توپ وجود دارد. حکم را با استقرا روی b ثابت می‌کنیم. برای پایه، حالت $b=n$ را در نظر می‌گیریم. برای هر نوع توپ، یک جعبه‌ی جدا در نظر می‌گیریم و یک توپ از آن نوع در جعبه‌ی مذکور می‌گذاریم. به این ترتیب آرایشی با $n = (b-n+1) \times n$ توپ ارائه می‌شود. حال فرض کنید حکم برای $b=k$ برقرار باشد. ثابت می‌کنیم حکم برای $b=k+1$ نیز برقرار است. یک جعبه را کنار می‌گذاریم و در k جعبه‌ی باقی‌مانده، آرایشی با $(k-n+1) \times n$ توپ، مطابق فرض استقرا ارائه می‌کنیم. حال در جعبه‌ی کنار گذاشته شده از هر نوع توپ، یکی می‌گذاریم. به این ترتیب یک آرایش مطلوب به دست می‌آید که $(k-n+2) \times n$ توپ دارد و حکم ثابت می‌شود. ■

بمب‌گذاری واس اوناس! .... ۶۵ امتیاز

الف) ثابت می‌کنیم پاسخ مسئله برابر با

$$\lfloor max_{1\leq i\leq k} lg(a_k+1)\rfloor$$

است.

□ ابتدا ثابت می‌کنیم، اگر شکل گراف شهرهای یک کشور، یک مسیر به ترتیب با شهرهای $v_1, v_2, \ldots, v_p$ باشد، حداقل $\lfloor lg(p) \rfloor$ مرحله، لازم است. برای این کار، کافی است ثابت کنیم اگر $p \geq 2^q$ باشد، حداقل q مرحله، لازم است. از استقرای قوی روی p استفاده می‌کنیم. برای پایه‌ی استقرا، $p=1$ را در نظر می‌گیریم. داریم $p \geq 2^0$ و تعداد مراحل لازم برای انجام کار، ۰ است. فرض کنید حکم برای $p >$ برقرار باشد. ثابت می‌کنیم حکم برای p نیز برقرار است. فرض کنید در مرحله‌ی ابتدایی، شهر v_i را انتخاب کنیم. اگر $i \leq 2^{q-1}$ باشد؛ ممکن است دستگاه شهر v_{i+1} را نشان دهد، متوجه می‌شویم بمب در یکی از شهرهای $v_{i+1}, v_{i+2}, \ldots, v_p$ است و طبق فرض استقرا دست کم $q-1$ مرحله لازم است (زیرا $p-i \geq 2^{q-1}$). پس با یک مرحله‌ی ابتدایی، دست کم q مرحله لازم است و حکم ثابت می‌شود. در حالتی که $i > 2^{q-1}$ نیز به طریق مشابه اثبات انجام می‌شود. ■

□ به روشی مشابه روند بالا، می‌توان به راحتی ثابت کرد برای یک مسیر p رأسی، $\lfloor lg(p) \rfloor$ مرحله کافی نیز هست (روش جست‌وجوی دودویی). ■

□ حال فرض کنید شکل گراف شهرهای یک کشور، به صورت یک دور به طور p باشد. هر رأسی در ابتدا انتخاب شود، نیمی از رأس‌ها حذف می‌شوند و تنها یک مسیر به طور $\lfloor \frac{p}{2} \rfloor$ باقی می‌ماند. انتخاب هر شهر دیگر به جز این مسیر، اطلاعاتی در مورد شهر بمب‌گذاری شده به ما اضافه نمی‌کند و یک مسیر باقی می‌ماند. پس طبق قسمت قبل، برای یک دور به طول p، کمینه‌ی تعداد مراحل برابر $\lfloor lg(p) \rfloor = \lfloor lg(\frac{p}{2}) \rfloor + 1$ است. ■

□ حال ثابت می‌کنیم علی‌رضا و فرهاد، دست کم به

$$\lfloor max_{1 \leq i \leq k} lg(a_k + 1) \rfloor$$

مرحله در کشور واس‌ماس نیاز دارند. بدون از دست دادن کلیت مسئله فرض کنید $a_1 \geq a_2, a_3, \ldots, a_k$ باشد. فرض کنید بمب در یکی از شهرهای استان ۱ باشد و علی‌رضا و فرهاد این اطلاعات اضافه را از ابتدا بدانند. انتخاب هر شهر جز پایتخت و شهرهای استان ۱، پاسخ مشخصی دارد و اطلاعات جدیدی اضافه نمی‌کند. پس یک دور باقی می‌ماند و فرهاد و علی‌رضا طبق قسمت قبل حداقل به

$$\lfloor lg(a_1 + 1) \rfloor = \lfloor max_{1 \leq i \leq k} lg(a_k + 1) \rfloor$$

مرحله نیاز دارند. ■

□ پس فقط کافی است ثابت کنیم

$$\lfloor max_{1 \leq i \leq k} lg(a_k + 1) \rfloor$$

مرحله برای پیدا کردن بمب کافی است. اگر فرهاد و علی‌رضا در ابتدا پایتخت را انتخاب کنند و پایتخت یکی از شهرهای استان i را به آن‌ها نشان بدهد (بدون از دست دادن کلیت مسئله فرض کنید شهر شماره ۱ از این استان)، آنگاه بمب در یکی از شهرهای $1, 2, \ldots, a_{\lfloor \frac{a_i+1}{2} \rfloor}$ است و طبق قسمت‌های گفته شده می‌توان با حداکثر $\lfloor max_{1 \leq i \leq k} lg(\frac{a_k+1}{2}) \rfloor$ مرحله بمب را پیدا کرد. با احتساب یک مرحله‌ی اولیه، می‌توان نتیجه گرفت با حداکثر

$$\lfloor max_{1 \leq i \leq k} lg(a_k + 1) \rfloor$$

می‌توان بمب را پیدا کرد. ■

ب) ثابت می‌کنیم پاسخ برابر r است.

□ ابتدا ثابت می‌کنیم r مرحله برای پیدا کردن بمب، کافی است. آن شهری که کم‌ترین عدد نسبت داده شده را دارد، در ابتدا انتخاب می‌کنیم. سپس در هر مرحله همان شهری را انتخاب می‌کنیم که دست‌گاه به ما نشان می‌دهد. به این ترتیب پس از حداکثر r مرحله، شهر بمب‌گذاری شده پیدا می‌شود (توجه کنید اگر rـامین مرحله انجام شد و هنوز بمب پیدا نشده بود، بمب حتمن در شهر بعدی است و نیازی به چک کردن آن نیست). ■

□ حال ثابت می‌کنیم نقشه‌ای با کمینه‌ی عدد r وجود دارد که حداقل r مرحله نیاز دارد. یک درخت دودویی کامل با ارتفاع r در نظر بگیرید.

با استقرا روی r ثابت می‌کنیم برای پیدا کردن بمب، حداقل r مرحله لازم است. برای پایه‌ی استقرا $r = 0$ (درخت تک‌رأسی) را در نظر می‌گیریم و حکم واضح است. فرض کنید حکم برای $r - 1$ برقرار باشد؛ ثابت می‌کنیم حکم برای r نیز برقرار است.

- اگر در ابتدا ریشه‌ی درخت انتخاب شود، زیردرختی که بمب در آن است، مشخص می‌شود. انتخاب هر شهر از زیردرخت دیگر در ادامه، اطلاعاتی اضافه نمی‌کند؛ زیرا پاسخ آن مشخص است. پس طبق فرض استقرا در ادامه حداقل به $r - 1$ مرحله نیاز داریم و حکم ثابت می‌شود.
- اما اگر در ابتدا ریشه را انتخاب نکنیم، ممکن است بمب در زیردرخت دیگر باشد. فرض کنید علاوه بر این که بمب، یک شهر را نشان می‌دهد، پس از این انتخاب، این اطلاعات اضافی نیز به علی‌رضا و فرهاد داده شود که بمب در زیردرخت دیگر است. پس انتخاب هر شهر جز زیردرخت دیگر، اطلاعاتی به آن دو اضافه نمی‌کند و پاسخ آن مشخص است. پس طبق فرض استقرا باز هم در ادامه نیاز به حداقل $r - 1$ مرحله داریم و حکم ثابت می‌شود.

پس ثابت کردیم گرافی داریم که در آن حداقل r مرحله لازم است. ■

پس پاسخ برابر r است.

ج) در هر مرحله، مجموعه‌ی تمام رأس‌هایی که ممکن است بمب در آن‌ها باشد را S در نظر می‌گیریم. واضح است که در ابتدا S مجموعه‌ی تمام شهرهاست. در هر مرحله، آن شهری از S را در نظر می‌گیریم که مجموع فواصلش از دیگر شهرهای درون S، کمینه باشد. این رأس را $v(S)$ می‌نامیم. حال ثابت می‌کنیم با پاسخی که دست‌گاه به ما می‌دهد، می‌توانیم نتیجه بگیریم حداقل نیمی از شهرهای S، نمی‌توانند بمب داشته باشند. به این ترتیب در هر مرحله S حداقل نصف می‌شود و حکم ثابت خواهد شد.

برهان خلف می‌زنیم. فرض کنید یکی از شهرهای مجاور $v(S)$ مانند u انتخاب شود و تعداد شهرهای نامزد برای بمب داشتن، بیش از نصف S بماند. پس u به ازای بیش از $\frac{|S|}{2}$ شهر، فاصله‌ی کم‌تری از آن شهرها نسبت به $v(S)$ دارد و فاصله‌ی دیگر شهرهای S نیز از u، حداکثر یکی بیش‌تر از فاصله‌ی‌شان از $v(S)$ است. پس مجموع فواصل u از بقیه‌ی شهرهای S در این مرحله کم‌تر بوده است و به تناقض می‌رسیم. پس فرض خلف باطل است و حکم ثابت می‌شود.