

Дизајн играчака

Име задатка	Дизајн играчака
Улаз	Интерактивни задатак
Излаз	Интерактивни задатак
Временско ограничење	1 секунда
Меморијско ограничење	256 мегабајт

Ви радите за фирму која дизајнира играчке. Прављење нове играчке функционише на следећи начин: Постоји n штапова, нумерисаних од 1 до n , који виरे из кутије. Неки парови штапова су повезани жицом унутар кутије. (Другим речима, штапови и жице формирају неусмерен граф, где су штапови чворови, а жице гране.) Жице нису видљиве изван кутије, и једини начин да сазнамо нешто о њима је да користимо **тестера** на паровима штапова: Можемо изабрати два штапа i и j таква да $i \neq j$, и тестирати да ли су два штапа повезана унутар кутије, директно или индиректно. (Тј, тестер одговори да ли постоји пут између два чвора у графу.)

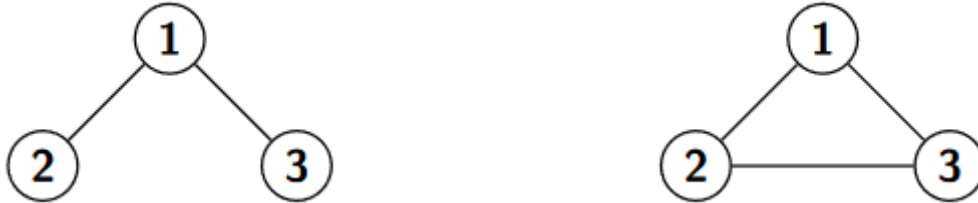
Овај скуп веза унутар кутије зовемо **дизајном** играчке.

Користите специјализовану апликацију да би тестирали и креирали ове дизајне. Апликација ради на следећи начин: Почиње са неким дизајном играчке, који називамо "дизајном 0". Апликација не приказује саме везе унутар кутије за овај дизајн. Уместо тога, можете поновити неколико пута следећу операцију, која се састоји од 3 корака:

1. Изаберете број дизајна a и два штапа са бројевима i и j таква да $i \neq j$.
2. Апликација вам каже шта би се десило ако би користили тестера на ова два штапа. Другим речима, каже вам да ли су штапови i и j (директно или индиректно) повезани у дизајну a .
3. Такође, ако штапови нису били директно или индиректно повезани у дизајну a , онда апликација креира нови дизајн који има све везе из дизајна a и још једну нову директну везу између штапова i и j . Овом дизајну је дат наредни доступни број. (На тај начин, први дизајн који је креиран ће имати број 1, наредни ће имати број 2, и тако даље.) Приметите да ово не мења дизајн a , само прави нови дизајн са једном додатом везом.

Ваш циљ је да сазнате што је више могуће о дизајну 0 применом ове операције

Приметите да није увек могуће одредити тачан скуп веза за дизајн 0, зато што не постоји начин да се разликују директне и индиректне везе. На пример, размотрите наредна два дизајна за $n = 3$:



Тестер ће за било који пар штапова рећи да је повезан у оба дизајна изнад, те према томе нећемо моћи да разликујемо два дизајна.

Ваш циљ је да одредите било који дизајн који је еквивалентан дизајну 0. Два дизајна су **еквивалентна** ако тестер пријави исти резултат за оба дизајна за све парове штапова.

Имплементација

Ово је *интерактивни задатак*. Морате имплементирати функцију

```
void ToyDesign(int n, int max_ops);
```

која одређује дизајн који је *еквивалентан* дизајну 0. Ваша имплементација би требала да оствари овај циљ позивањем ове две функције на начин који је описан испод. Прва функција коју можете да позовете:

```
int Connected(int a, int i, int j);
```

где је $1 \leq i, j \leq n, i \neq j, a \geq 0$, и a не сме да буде веће од броја дизајна до сада креираних од стране апликације. Ако су штапови i и j (директно или индиректно) повезани у дизајну a , онда ће функција вратити вредност a . У супротном, повратна вредност функције ће бити број дизајна који су направљени до сада плус један, што постаје број који је додељен новом дизајну који садржи све вете из дизајна a и нову везу између i и j . Функција `Connected` може бити позвана највише `max_ops` пута.

Када ваш програм заврши са примењивањем операције `Connected`, он такође треба да опише дизајн који је еквивалентан дизајну 0. Да би описао дизајн, програм би требао да позове функцију:

```
void DescribeDesign(std::vector<std::pair<int, int>> result);
```

Аргумент `result` је вектор парова целих бројева који описује директне везе између штапова. Сваки пар одговара једној вези и треба да садржи индексе два штапа који су повезани жицом. Може бити највише једна директна веза између сваког (неуређеног) пара штапова, и не сме бити директних веза између неког штапа и самог себе. Позивање ове функције окончава извршавање програма.

Ограничења

- $2 \leq n \leq 200$

Scoring

- Подзадатак 1 (10 поена): $n \leq 200$, $max_ops = 20\,000$
- Подзадатак 2 (20 поена): $n \leq 8$, $max_ops = 20$
- Подзадатак 3 (35 поена): $n \leq 200$, $max_ops = 2\,000$
- Подзадатак 4 (35 поена): $n \leq 200$, $max_ops = 1\,350$

Пример интеракције

Радња такмичара	Радња грејдера	Објашњење
	<code>ToyDesign(4, 20)</code>	Играчка садржи 4 штапа. Треба да одредите било који дизајн који је еквивалентан дизајну 0 позивањем функције <code>Connected</code> највише 20 пута.
<code>Connected(0, 1, 2)</code>	Враћа вредност 1.	Штапови 1 и 2 нису повезани директно или индиректно у дизајну 0. Сада је направљен дизајн 1.
<code>Connected(1, 3, 2)</code>	Враћа вредност 2.	Штапови 3 и 2 нису повезани директно или индиректно у дизајну 1. Нови дизајн 2 је сада направљен.
<code>Connected(0, 3, 4)</code>	Враћа вредност 0.	Штапови 3 и 4 су повезани директно или индиректно у дизајну 0. Не прави се нови дизајн.
<code>DescribeDesign({{3, 4}})</code>	-	Описујемо дизајн који има само једну везу: између

Пример грејдера

Пример грејдера који је на располагању, `grader.cpp`, у архиви `ToyDesign.zip`, чита са стандардног улаза у следећем формату:

- Прва линија садржи број штапова, n , број директних веза, m и max_ops
- Наредних m линија садрже директне везе описане паровима штапова.

Пример грејдера чита са стандардног улаза и позива функцију `ToyDesign` из решења корисника. Грејдер ће исписати једну од наредних порука, на основу вашег решења:

- "Wrong answer: Number of operations exceeds the limit.", ако је број позива функције `Connected` већи max_ops
- "Wrong answer: Wrong design id.", ако је параметар a у позиву функције `Connected` број дизајна који не постоји у моменту позива функције.
- "Wrong answer: Incorrect design.", ако дизајн описан у `DescribeDesign` није еквивалентан дизајну 0.
- "OK!" ако је дизајн описан у `DescribeDesign` еквивалентан дизајну 0.

Да би компајлирали дати грејдер заједно са вашим решењем, можете користити наредну команду у терминалу:

```
g++ -std=gnu++11 -O2 -o solution grader.cpp solution.cpp
```

где је `solution.cpp` ваше решење које шаљете на CMS. Да би извршили програм са примером улаза који је дат у прилогу, откуцајте следећу команду у терминалу:

```
./solution < input.txt
```