

Државно такмичење из програмирања Београд – 9. мај 2010. II категорија (7. и 8. разред)

Шибице

Игра коју су, ипак, најчешће играли била је "шибицарење". У оном истом ресторану пронашли су M кутија шиблица, а од папирића су направили једну лоптицу. Игра се састојала у следећем: кутије се поређају једна поред друге у линији, па се испод прве постави лоптица. Један од играча N пута ($N \leq 10\,000$) мења места двама случајно одабраним кутијама, а на сваких K замена избацује једну кутију по избору, при чему не сме остати мање од 3 кутије (због чега се игра и не започиње уколико не постоји довољан број кутија, тј. ако је број избацивања превелик). Након свих N замена други играч треба да погоди испод које кутије се налази лоптица.

Ваш задатак је да напишете програм **SIBICE** који за дати број кутија M ($M \leq 10\,000$), број замена N , а затим и за дате парове замена и редне бројеве кутија за избацивање исписује:

- позисију лоптице у низу кутија, након извршавања свих N замена и одговарајућег броја избацивања,
- 1 уколико је лоптица избачена током игре и то у тренутку када је избачена (не учитавају се даљи потези),
- 2 уколико нема довољно кутија, што се може десити када игра довољно дуго траје да се избацивањима кутија избаце све пре краја (с обзиром да су N , M и K познати пре почетка игре није потребно учитавати предвиђене потезе, већ исписати -2 пре тога).

Улазни подаци. Стандардни улаз садржи више линија:

- прва – три цела броја M , N и K , где је M број кутија, N број замена, K број који представља након колико замена се обавља избацивање једне кутије,
- све остале линије, у зависности од тога да ли је на раду замена или избацивање, садрже:
 - два цела броја који представљају тренутне позиције кутија којима би требало заменити места (подразумева се да позиције постоје),
 - један цео број који представља редни број кутије коју треба избацити (подразумева се да позиција постоји).

Пример.

Улаз:	Улаз:
5 4 2	5 15 2
1 2	Изназ:
2 4	-2
3	
3 2	
3 4	
4	
Изназ:	
2	

Изназни подаци. Једина линија стандардног излаза садржи један цео број, чије је значење описано у претходном тексту.

Понуда

Омиљене игре свих млађих путника у возу биле су игрице на конзоли. С обзиром да је почетком пута избила свађа око тога ко ће и колико играти, неко од одраслих решио је да приступ конзоли ограничи увођењем својеврсних квалификација. Квалификација се састојала у игри Најмања једниствену понуда. Награда за победника ове игре била је полчасовни тенис меч на конзоли за игрице. Игра је била прилично једноставна, свако од играча је могао да на папиру (тајно) да предлоге колико год жели (позитивних целих) бројева, тзв. понуда. Један од одраслих је сакупио понуде свих играча и победник је био онај који је предложио најмањи број, који нико други није предложио, тј. који је дао најмању јединствену понуду. Дешавало се да ни једна од понуда није једниствену.

Ваш задатак је да напишете програм **PONUДА** који за дате понуде свих играча одређује која је најмања једниствену, ако постоји, а ако не постоји дате вашу, али тако да она буде већа од најмањег понуђеног броја.

Улазни подаци. У првој линији улаза се налази цео број ($N \leq 1\,000$) који представља укупан број датих понуда, а у следећих N линија се налази по један позитиван цео број који представља једну од понуда.

Изназни подаци. Излаз садржи један позитиван цео број који представља већ дату понуду, а уколико понуда није дата, вашу понуду (већу од најмањег понуђеног).

Пример.

Улаз:	Изназ:	Улаз:	Изназ:
12	5	7	13
4		10	
4		11	
1		10	
3		12	
1		11	
3		12	
2		15	
2			
2			
6			
5			
6			

Линије

Још пре пута, Огњен и Вук су се договорили које ће све локације у Москви посетити. Гледајући градске аутобуске линије, направили су списак оних које имају бар једну станицу на локацији која је њима интересантна, али тако да ако би се провезли свим линијама са списка сигурно би могли да посете све планиране локације. Локација је било N , као и аутобуских линија које су одабрали. Током пута су проучавали свој списак и увидели да су неке од линија сувишне. Одлучили су да их избришу са списка, не би ли смањили трошкове превоза, а да при томе, ипак, успеју да обиђу све што су планирали. Критеријум по коме су избацивали линије, водећи рачуна о томе да све локације остану покривене, био је следећи:

- прво су избацивали линије које су покривале **највећи број локација** (јер су захтевале више чекања. Наиме, на једној линији сиђу на првој локацији, па кад заврше обилазак морају да чекају следећи превоз на истој линији не би ли стигли истим путем до следеће локације);
- ако су две линије покривале исти (највећи) број локација, онда су избацивали ону која садржи локацију са **(нај)већом покривеношћу** свим линијама (пример. линија 1: локација 3 (покрива је 5 линија), локација 5 (покрива је 3 линије); линија 2: локација 2 (покрива је 3 линије), локација 4 (покрива је 3 линије) – у овом случају би линија 1 била избрисана због локације 3);
- ако су две линије по броју локација и локацијама са највећом покривеношћу биле подједнаки кандидати за брисање, била би избачена линија чија је локација са **минималном покривеношћу** била посећенија од локације са минималном покривеношћу других линија (пример. линија 1: локација 3 (покрива је 5 линија), локација 5 (покрива је 3 линије); линија 2: локација 2 (покрива је 5 линија), локација 4 (покрива је 2 линије) – у овом случају би линија 1 била избрисана због локације 5);
- ако су две линије по броју локација, локацијама са највећом покривеношћу и минималном покривеношћу биле подједнаки кандидати за брисање, била је избачена она која је на списак била **раније уписана**.

Ваш задатак је да напишете програм **LINIJЕ** који би могао да обави посао уместо Огњена и Вука, тј. да утврди колико је највише линија које је могуће избацити, а да при томе све локације остану покривене.

Прва линија улаза садржи један цео број, који представља број линија/локација. Наредних N ($N \leq 100$) линија садржи по неколико целих бројева, при чему, i -та линија садржи број локација које i -та аутобуска линија покрива, а затим и редне бројеве тих локација.

Излаз садржи број K , који представља максималан број линија које је могуће избацити према претходно описаном поступку.

Улазни подаци. Прва линија улаза садржи један цео број, који представља број линија/локација. Наредних N ($N \leq 100$) линија садржи по неколико целих бројева, при чему, i -та линија садржи број локација које i -та аутобуска линија покрива, а затим и редне бројеве тих локација.

Излазни подаци. Излаз садржи број K , који представља максималан број линија које је могуће избацити према претходно описаном поступку.

Пример.

Улаз:	Излаз:	Коментар на пример:
5	2	Биће избачене 2. па 1. линија.
4 1 2 3 4		
4 5 3 2 1		
3 1 2 5		
3 2 5 3		
2 3 4		
<hr/>		
Улаз:	Излаз:	Коментар на пример:
3	0	Није могуће избацити било коју линију.
1 1		
2 2 3		
1 3		