

"Tree Search" (Претрага Стабла)

Дато је укореењено бинарно стабло са N чворова. Чворови су нумерисани бројевима 1 до N , корен је чвор број 1. Сваки чвор има једног родитеља у стаблу. Стабло је бинарно, тј. сваки чвор може бити родитељ највише два чвора.

Један од чворова је посебан. Ви покушавате да га пронађете. Можете постављати питања следеће врсте: "Да ли је посебни чвор у подстаблу чвора x "? Чвор y је у подстаблу чвора x ако и само ако најкраћи пут од чвора y до корена (чвора 1) пролази кроз x . Чвор x такође припада подстаблу чвора чвора x .

Можете постављати ово питање највише 35 пута. Након тога треба да одговорите који је чвор посебан.

Детаљи имплементације

Потребно је да имплементирате наредну функцију:

```
int solve(int N, std::vector < int > p)
```

- N : број чворова
- p низ од тачно $N - 1$ елемената који описују стабло: чвор $p[i]$ (где $1 \leq p[i] \leq i + 1$) је родитељ чвора $i + 2$ за свако $0 \leq i \leq N - 2$
- Ни један елемент у p се не појављује више од два пута
- Ова функција треба да врати број посебног чвора
- Ова функција се позива тачно једном

Ваша функција може да позива следећу функцију:

```
int ask(int x)
```

- x : број чвора
- $1 \leq x \leq N$
- Функција враћа 1 ако је специјални чвор у подстаблу чвора x , а 0 иначе.

Ограничења

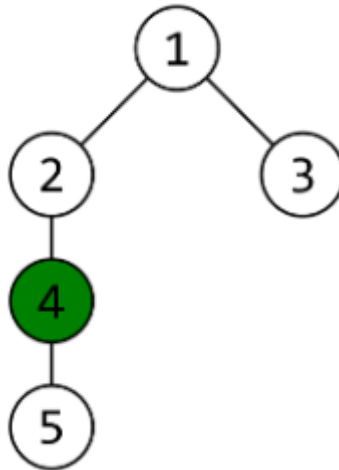
- $2 \leq N \leq 100\,000$

Пример

Разматрајмо наредни позив:

```
solve(5, [1, 1, 2, 4])
```

Стабло је сачињено од грана (1,2), (1,3), (2,4) и (4,5).



Ваш програм је урадио следећи позив:

```
ask(4)
```

што је вратило 1. Након тога је ваш програм направио позив:

```
ask(5)
```

што је вратило 0.

Ваш програм је закључио да је чвор 4 посебан и вратио 4.

Ограничења

- $2 \leq N \leq 100\,000$

Подзадачи

1. (20 поена) $N \leq 35$
2. (30 поена) $p[i] = i + 1$ за свако $0 \leq i \leq N - 2$
3. (15 поена) $p[i] = \lfloor i/2 \rfloor + 1$ за свако $0 \leq i \leq N - 2$
4. (35 поена) Нема додатних ограничења.

Грејдер за примере

Грејдер прима улаз у следећем формату:

- линија 1: N
- линија 2: $p[0], p[1], \dots, p[N - 2]$

Грејдер исписује све упите у следећем формату:

- линија 1: $? x$

Грејдер чита одговоре на упите у следећем формату:

- линија 1: y

Грејдер исписује ваша погађања у следећем формату:

- линија 1: $! x$

"Teleporters" (Телепортери)

Ана и Бека су на различитим тачкама координатне линије и планирају да се сретну. Једини начин кретања је коришћење телепортера.

Постоји N телепортера, i -ти телепортер се налази на $c[i]$ координати и ради на фреквенцији $f[i]$. Ипак, нису сви тренутно доступни; само оне чије су фреквенције у интервалу $[L, R]$ могу бити коришћене.

Коришћење телепорта траје минут и телепортује корисника на координату која је симетрична почетној у односу на позицију телепортера. Другим речима, ако је почетна координата x_1 , након коришћења телепортера i , крајња координата x_2 задовољиће једначину $(x_1 + x_2)/2 = c[i]$.

Сваког минута, Ана и Бека морају да користе један од слободних телепортера (не нужно различита). Оне ће комуницирати током телепортације, и доживеће неудобност једнаку апсолутној разлици фреквенција њихових телепортера које користе. Тежина читавог путовања је дефинисана као максимална неудобност коју су доживеле.

Дато је Q различитих сценарија, и за сваки, твој задатак је да одредиш да ли ће се Ана и Бека икада срести коришћењем доступних телепортова, и ако се то деси, која је минимална могућа тежина путовања за њих.

Један сценарио одређен је са четири цела броја:

- A : Анина почетна координата
- B : Бекина почетна координата
- L : Минимална фреквенција доступних телепортова
- R : Максимална фреквенција доступних телепортова.

За сваки сценарио, испиши минималну тежину путовања ако је сусрет могућ и -1 иначе. Укупно време путовања није битно за израду овог задатка.

Формат улаза

У првој линији се учитавају два цела броја : N и Q .

У другој линији се учитава N целих бројева: $c[1], c[2], \dots, c[N]$.

У трећој линији се учитава N целих бројева: $f[1], f[2], \dots, f[N]$.

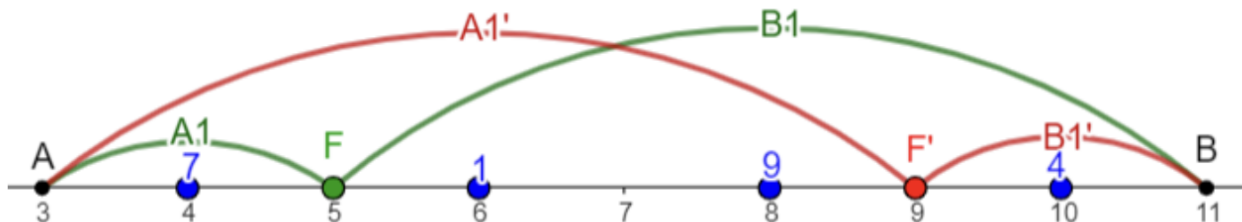
У свакој од наредних Q линија описан је сценарио са четири цела броја: A, B, L и R ($A \neq B$).

Формат излаза

Исписати Q целих бројева раздвојених бланко карактером у једној линији: одговоре на сценарије 1, 2, ..., Q .

Пример 1

Стандардни улаз	Стандардни излаз
4 3	2 3 -1
4 6 8 10	
7 1 9 4	
3 11 1 50	
3 11 1 5	
5 7 1 1	



У првом сценарију, ако Ана користи телепортер 2 и Бека користи телепортер 4, сретће се на координати 9 са неудобношћу $|1 - 4| = 3$.

Боље решење је да Ана користи телепортер 1 и Бека користи телепортер 3; у овом случају, сретће се на $F' = 5$ и искусиће неудобност $|7 - 9| = 2$.

У другом сценарију, боља опција није више доступна због ограниченог интервала фреквенције.

У трећем сценарију, доступан је само један телепортер, и сусрет није могућ.

Пример 2

Стандардни улаз	Стандардни излаз
3 3	-1 2 7
-2 1 -1	
10 1 3	
-6 6 20 20	
-6 6 0 20	
-6 6 2 20	

Координате могу бити негативне.

Ограничења

- $2 \leq N \leq 50\,000$
- $1 \leq Q \leq 50\,000$
- $1 \leq f[i] \leq 10^9$
- $-10^9 \leq c[i], A, B \leq 10^9$
- $1 \leq L \leq R \leq 10^9$

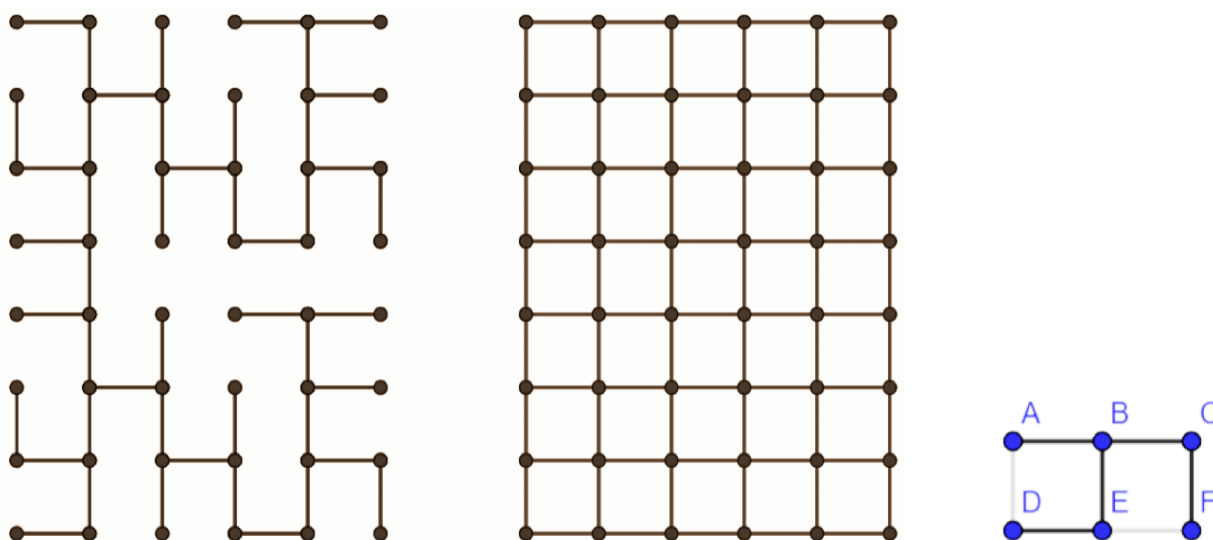
Подзадаци

1. (11 поена) $N, Q \leq 10$; $|c[i]|, f[i] \leq 50$ за сваки $1 \leq i \leq N$.
2. (10 поена) $N \leq 100$; $L = 1$; $R = 10^9$; $|c[i]|, f[i] \leq 100$ за сваки $1 \leq i \leq N$.
3. (5 поена) $N = 2$; $L = 1$; $R = 10^9$
4. (9 поена) $N \leq 1\,000$; $L = 1$; $R = 10^9$; $f[i] = 1$ за сваки $1 \leq i \leq N$.
5. (6 поена) $L = 1$; $R = 10^9$; $f[i] = 1$ за сваки $1 \leq i \leq N$.
6. (7 поена) $N \leq 1\,000$; $L = 1$; $R = 10^9$
7. (17 поена) $L = 1$; $R = 10^9$
8. (8 поена) $L = 1$
9. (14 поена) $N, Q \leq 20\,000$
10. (13 поена) Без додатних ограничења.

"Opening Offices" (Отварање Канцеларија)

Ваша компанија планира да отвори своје канцеларије у граду са N хоризонталних и M вертикалних улица са зградама на сваком пресеку улица. Свака зграда је повезана са свим својим суседима са највише две хоризонталне и две вертикалне улице, свака дужине 1.

Ноћу, је само $N \times M - 1$ путева осветљено, а остали се не могу користити. Ови путеви су такви да формирају стабло, тј. тачно су такви да су сваке две зграде повезане.



Прва слика приказује улице ноћу, док друга приказује исте улице дању. Трећа слика је једноставнији пример који ће бити коришћен у примерима испод.

Сваку зграду је могуће купити и ту направити канцеларију. Сваког месеца ћете обићи све канцеларије, почевши од једне канцеларије, обилазећи све остале канцеларије и на крају се вратити у почетну канцеларију. Приликом облиска канцеларија, користите доступне улице и минимизовати укупну дужину пута, иако нисте сигурни за тачно доба дана.

У примеру на десној страни, у случају да се канцеларије отворе у зградама A , D и F , укупна дужина пута би била 6 током дана, а 10 током ноћи.

Како бисте избегли компликације при планирању, одлучили сте да изаберете канцеларије тако да минимална дужина пута остане иста како током дана тако и током ноћи.

Потребно је да израчунате број начина на који се канцеларије могу изабрати тако да се задовољи дати услов. Два избора се сматрају различитим ако постоји бар једна зграда која је

присутна у једном избору, а није присутна у другом. Како број начина може бити велик, потребно је да га израчунате по модулу 1 000 000 007.

Приметите да постоји ограничење за број канцеларија. Погледајте формат улаза за детаље.

Формат улаза

Прва линија садржи три цела броја: N , M и T . T представља **тачан** број канцеларија које планирате да отворите, осим када је $T = 1$, у том случају можете отворити **било који број** канцеларија, али **барем две**.

Свака од наредних N линија садржи M карактера (без размака). j -ти карактер на $i + 1$ -ој линији је или '0', '1', '2' или '3', и описује путеве који су осветљени ноћу из зграде на i -тој улици са врха и j -тој улици са лева:

- '0' означава да нема путева од ове зграде до зграде изнад од ње и зграде лево од ње.
- '1' означава да постоји пут од ове зграде до зграде директно изнад ње.
- '2' означава да постоји пут од ове зграде до зграде директно лево од ње.
- '3' означава да постоје путеви од ове зграде до зграде директно изнад ње и до зграде лево од ње.

Има тачно $N \times M - 1$ путева и они чине стабло.

Формат излаза

Исписати један цео број: број начина по модулу $10^9 + 7$.

Пример 1

Стандардни улаз	Стандардни излаз
2 3 2	12
022	
031	

Овај пример одговара примеру са слике изнад.

Канцеларије се могу отворити у следећим паровима зграда: {A, B}, {A, C}, {A, E}, {A, F}, {B, C}, {B, D}, {B, E}, {B, F}, {C, D}, {C, E}, {C, F}, {D, E}.

Пример 2

Стандардни улаз	Стандардни излаз
2 3 3	10
022	
031	

Исти град при чему је $T = 3$. Канцеларије се могу отворити у следећим тројкама зграда: {A, B, C}, {A, B, E}, {A, B, F}, {A, C, E}, {A, C, F}, {B, C, D}, {B, C, E}, {B, C, F}, {B, D, E}, {C, D, E}.

Пример 3

Стандардни улаз	Стандардни излаз
2 3 1	25
022	
031	

Поред могућности за $T = 2$ и $T = 3$ приказане у примерима изнад, канцеларије се могу отворити и на следеће начине: {A, B, C, E}, {A, B, C, F}, {B, C, D, E}.

Ограничења

- $1 \leq T \leq 3$
- $1 \leq N, M \leq 1\,000$

Подзадаци

1. (4 поена) $M, N \leq 2$
2. (5 поена) $N = 1$
3. (9 поена) $T = 2; N, M \leq 50$
4. (11 поена) $T = 2$
5. (9 поена) $T = 3; N, M \leq 20$
6. (13 поена) $T = 3$
7. (14 поена) $T = 1; M, N \leq 4$
8. (10 поена) $T = 1; N, M \leq 50$
9. (9 поена) $T = 1$; Описи путева не садрже карактер '3'.
10. (16 поена) $T = 1$

"Square Grid Puzzle" (Пазл са квадратићима)

У овом пазлу је дата $N \times N$ квадратна табла са бројевима од 0 до $N \times N - 1$, укључујући 0 и $N \times N - 1$. Циљ је да се добије табла у којој је број на пресеку i -те врсте и j -те колоне једнак $i \times N + j$ за свако $0 \leq i, j < N$. Да бисте постигли циљ, можете користити два типа потеза:

- **Down move** (потез доле): "**D** $a[0]$ $a[1]$... $a[N - 1]$ ", где је $a[0]$, $a[1]$, ... , $a[N - 1]$ нека пермутација бројева из најгорње врсте табле. Овим потезом се уклања најгорња врста и додаје се нова врста са бројевима $a[0]$, $a[1]$, ... , $a[N - 1]$ (са лева на десно) на дно табле.
- **Right move** (потез десно): "**R** $b[0]$ $b[1]$... $b[N - 1]$ ", где је $b[0]$, $b[1]$, ... , $b[N - 1]$ нека пермутација бројева из најлевље колоне табле. Овим потезом се уклања најлевља колона и додаје се нова колона са бројевима $b[0]$, $b[1]$, ... , $b[N - 1]$ (од горе на доле) као најдеснија колона табле.

Пермутација реда односно колоне је промена редоследа бројева без додавања или уклањања бројева. Редослед може остати и непромењен.

На пример, ако је тренутна табла:

Ред/Колона	0	1	2
0	2	4	6
1	8	1	5
2	7	3	0

Извршавањем потеза "**D** 6 2 4", добијамо следећу таблу:

Ред/Колона	0	1	2
0	8	1	5
1	7	3	0
2	6	2	4

Међутим, ако извршимо потез "**R** 2 8 7", добијамо:

Ред/Колона	0	1	2
0	4	6	2
1	1	5	8
2	3	0	7

За $N = 3$, циљна табла изгледа овако:

Ред/Колона	0	1	2
0	0	1	2
1	3	4	5
2	6	7	8

Желите да решите пазл у мање од $3 \times N$ потеза. Међутим, могуће је добити парцијалне поене ако користите више потеза или ако не успете да решите пазл. Погледати секцију "бодовање" за детаље.

Формат улаза

Прва линија садржи један цео број: N .

Наредних N линија описује почетну таблу, са N бројева у свакој линији.

Формат излаза

Прва линија треба да садржи један цео број: M , број потеза.

Свака од наредних M линија треба да садржи један потез.

Бодовање

Означимо са M број потеза у вашем решењу. Додатно, дефинишимо $A = 3 \times N$ и $B = 2 \times N^2$.

Ако ваш излаз није валидан или је $M > B$, добијате 0 поена. Иначе ваш број поена зависи од тога колико бројева имате на тачним позицијама (означимо то са C).

Ако је $C < N \times N$, пазл није решен и добићете $(50 \times \frac{C}{N \times N})\%$ поена за тај тест пример. Иначе:

- Ако је $M < A$ добијате 100% поена за тај тест пример.
- Ако је $A \leq M \leq B$ добићете $(40 \times (\frac{B-M}{B-A})^2 + 50)\%$ поена за тај тест пример.

Сваки тест пример вреди исти број поена. Ваш крајњи број поена је збир поена по тест примерима. За крајњи скор се гледа број поена на најбољем сабмишну.

Пример 1

Стандардни улаз	Стандардни излаз
3	4
1 4 2	R 3 6 1
3 7 5	D 2 3 4
6 8 0	D 5 6 7
	R 2 5 8

Ово решење достиже циљну таблу у мање од 9 потеза, па добија све поене.

Пример 2

Стандардни улаз	Стандардни излаз
2	0
2 1	
0 3	

Пазл није решен јер се само два броја (1 и 3) од 4 налазе на тачним позицијама. Ово решење би добило $50 \times \frac{2}{4} = 25\%$ поена за тај тест пример.

Ограничења

- $2 \leq N \leq 9$

Подзадаци

- Нема подзадатака.
- Број тест примера за свако N од 2 до 9 је исти.

"Tree Infection" (Инфицирано дрво)

Дато је укореењено дрво које садржи N чворова, заједно са целим бројевима R и M . Чворови су нумерисани од 1 до N , чвор 1 је корен. Сваки чвор има једног родитеља.

Ако је чвор s изабран, постаје инфициран као и сви његови потомци (односно сви чворови до којих се долази пратећи гране на доле од чвора s) **на растојању R или мањем**. Раздаљина између два чвора је број грана на путу између њих. Из чвора u се може доћи до чвора v само под условом да ниједан од њих није инфициран, и да број инфицираних чворова на путу између њих **је мањи или једнак M** .

За сваки чвор s ($1 \leq s \leq N$), морате израчунати број парова чворова (u, v) тако да је $1 \leq u < v \leq N$ и да се из u може доћи до v (и обрнуто).

Формат улаза

Прва линија садржи три цела броја: N , R и M .

Друга линија садржи $N - 1$ целих бројева: $p[2]$, $p[3]$, ..., $p[N]$, родитеље чворова 2, 3, ..., N , редом.

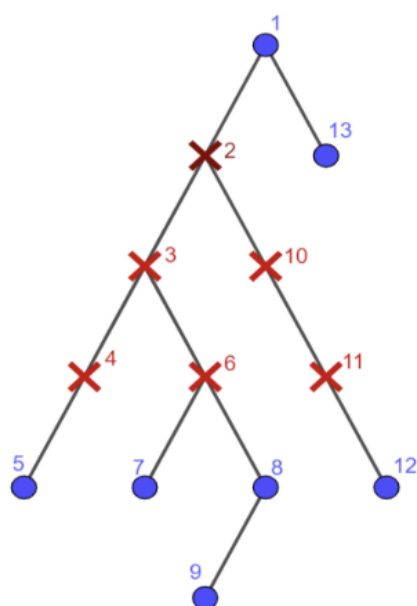
Формат излаза

Исписати N линија које садрже један цео број: s -та линија треба да садржи број парова у случају када је изабрани чвор s .

Не препоручује се коришћење `std::endl` за прелазак у нови ред. Користити `'\n'` за боље перформансе.

Пример 1

Стандардни улаз	Стандардни излаз
13 2 2	16
1 2 3 4 3 6 6 8 2 10 11 1	4
	15
	55
	66
	36
	66
	55
	66
	45
	55
	66
	66



Слика изнад одговара примеру $s = 2$.

Доступни парови: (1,13), (7,8), (7,9), (8,9).

Листа не укључује пар (1,2) јер је чвор 2 инфициран. Слично, пар (1,5) није на листи јер су три чвора између 1 и 5 инфицирана (2, 3 и 4).

Пример 2

Стандардни улаз	Стандардни излаз
3 0 1	1
1 2	1
	1

Ограничења

- $2 \leq N \leq 500\,000$
- $1 \leq p[i] < i$ (за сваки $2 \leq i \leq N$)
- $0 \leq R \leq N - 1$
- $0 \leq M \leq 2 \times R + 1$

Подзадаци

1. (20 поена) $N \leq 300$
2. (14 поена) $R = 0$
3. (15 поена) $M = 2 \times R + 1$
4. (10 поена) $M = 2 \times R - 1$
5. (16 поена) $N \leq 5\,000$
6. (25 поена) Без ограничења.

"Team Building" (Тимбилдинг)

Желите да окупите тим од N програмера. Већ сте истражили програмере и проценили да је ниво вештине i -те особе ($1 \leq i \leq N$) представљен целим ненегативним бројем $s[i]$. Схватили сте да је оно што је заиста важно редослед којим их запошљавате.

Сваки програмер је представљен са две додатне целобројне вредности: радна стопа и мотивација, од којих су обе једнаке 0 на почетку, али се могу повећати запошљавањем нових чланова тима. Када се запосли нови програмер, онда се појављује следећи списак догађаја задат наредним редоследом:

- Нови програмер се придружује тиму са радном стопом и мотивацијом иницијализованом на 0.
- Радна стопа претходно ангажованих програмера повећава се за њихову вредност мотивације.
- Мотивација раније ангажованог програмера повећава се нивоом вештине новозапослених.

Снага тима се потом утврђује збиром радних стопа свих чланова тима. Ваш циљ је да израчунате максималну могућу снагу тима оптимизацијом редоследа запошљавања.

На пример, ако ангажујете програмера чији ниво вештина је редом $(0, 2, 2, 3)$, процес запошљавања ће утицати на њихове вредности на следећи начин:

Догађај	Радна стопа	Мотивација
Запошљавање са вештином 0	0	0
Запошљавање са вештином 2	0 0	0 0
Ажурирање радних стопа	0 0	0 0
Ажурирање мотивација	0 0	2 0
Запошљавање са вештином 2	0 0 0	2 0 0
Ажурирање радних стопа	2 0 0	2 0 0
Ажурирање мотивација	2 0 0	4 2 0
Запошљавање са вештином 3	2 0 0 0	4 2 0 0
Ажурирање радних стопа	6 2 0 0	4 2 0 0
Ажурирање мотивација	6 2 0 0	7 5 3 0

Дакле, снага тима је једнака $6 + 2 + 0 + 0 = 8$. Међутим, ако ангажујете програмере у бољем редоследу $(2, 2, 3, 0)$, добићете снагу тима једнаку $7 + 3 + 0 + 0 = 10$.

Ново запошљавање	Радна стопа	Мотивација
2	0	0
2	0 0	2 0
3	2 0 0	5 3 0
0	7 3 0 0	5 3 0 0

Штавише, током наредних Q дана, добићете обавештења о променама у процени нивоа вештина одређених програмера.

После дана i , ниво вештине програмера $x[i]$ биће ажуриран у вредност $y[i]$ (што може да одговара претходној вредности). Ова ажурирана вредност вештине ће се користити у наредним данима, све док се потенцијално поново не ажурира.

Након сваког дана, почевши од данас, ваш циљ је да одредите максималну могућу снагу тима ангажовањем свих N програмера, узимајући у обзир процењене нивое вештина у том тренутку.

Формат улаза

Први ред садржи два цела броја: N и Q .

Други ред садржи целе бројеве: $s[1], s[2], \dots, s[N]$.

Затим, постоји Q редова, од којих i -ти ред садржи садржи два цела броја: $x[i]$ и $y[i]$ редом.

Формат излаза

Ваш програм мора да испише $Q + 1$ редова, тако да сваки ред садржи један цео број. Ови цели бројеви представљају највећу потенцијалну снагу тима након сваког дана, у хронолошком редоследу.

Пример

Стандардни улаз	Стандардни излаз
4 2	10
2 0 2 3	14
2 4	12
4 0	

После првог дана, нивои вештина ће бити ажурирани на $(2, 4, 2, 3)$ и максимална достижна снага тима постаје 14, а после другог дана ће бити додатно прилагођена вредност $(2, 4, 2, 0)$.

Ограничења

- $2 \leq N \leq 50\,000$
- $1 \leq Q \leq 100\,000$
- $0 \leq s[i] \leq 100\,000$ за свако $1 \leq i \leq N$.
- $1 \leq x[i] \leq N$ за свако $1 \leq i \leq Q$.
- $0 \leq y[i] \leq 100\,000$ за свако $1 \leq i \leq Q$.

Подзадаци

1. (11 поена) $N \leq 7$; $Q \leq 100$
2. (19 поена) $N, Q \leq 500$
3. (15 поена) $Q \leq 10$
4. (6 поена) Нивои вештина су мањи или једнаки 1.
5. (9 поена) Нивои вештина су мањи или једнаки 500.
6. (12 поена) $x[i] = 1$ за свако $1 \leq i \leq Q$.
7. (10 поена) Свако ажурирање ће променити ниво вештине за највише 1.
8. (18 поена) Нема додатних ограничења.