Magyar újforint

A nemzeti bank új bankjegyeket bocsát ki, melyeknek a neve Magyar újforint, rövidítve HUF. Mivel a bank elnökének az `1` a szerencseszáma, ezért minden címlet csak `1`-es számjegyekből áll, például `1`, `11`, `111`, `1111` és így tovább.

A bank szeretné igazolni, hogy a Magyar újforint egy jó pénznem, ezért egy olyan programra lenne szükségük, ami meg tudja mondani, hogy egy adott összeg legkevesebb hány darab Magyar újforintos bankjegyből állítható össze.

Írj programot, ami ellátja ezt a feladatot: egy összeget kap bemenetként, és megmondja a kifizetéséhez szükséges Magyar újforintos bankjegyek számát.

Bemenet

A `standard bemenet` első és egyetlen sorában a kifizetendő összeg található.

Kimenet

A `standard kimenetre` egyetlen sort kell írni a szükséges bankjegyek számával.

Példa

Bemenet	Kimenet
23	3

Az első példában egy darab `1`-es és két darab `11`-es bankjeggyel kifizethető a `23`-as összeg. Belátható, hogy háromnál kevesebb bankjegy nem elég.

Bemenet	Kimenet
12345	5

Bemenet	Kimenet
282828	28

Korlátok

$1 \leq N \leq 10^9$

Időlimit: 1.0 s

Memórialimit: 256 MB

Pontozás

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	$N \leq 1000$	40
2	$N \leq 1\,000\,000$	20
3	nincsenek további megkötések	40

Irkafirka

Hanga vázlatfüzetének egyik oldalán egy `N` × `M`-es négyzetrács található. Hanga a négyzetrács néhány cellájába beírta a kedvenc szavának betűit, mindegyiket pontosan egyszer. A többi cella üres maradt.

Azt mondta, hogy a kedvenc szavát **egyértelműen** ki lehet olvasni a négyzetrácsból a következő módon:

- A bal felső sarokból indulva, minden lépésben jobbra és lefelé haladva elmegyünk a jobb alsó sarokig úgy, hogy az összes betűt bejárjuk.
- A betűket a bejárás sorrendjében egymás után olvasva kapjuk meg a kedvenc szavát.

Írj programot, ami a négyzetrácsból kiolvassa a kedvenc szót.

Bemenet

A `standard bemenet` első sorában a négyzetrács sorainak `N` és oszlopainak `M` száma szerepel.

A következő `N` sor mindegyike `M` karakterből áll. Az `i`-edik sor `j`-edik karaktere a megfelelő cella tartalmát adja meg:

- az angol ábécé kisbetűi (`a-z`) a kedvenc szó egy-egy betűjét jelölik,
- a pont (`.`) egy üres cellát jelöl.

Kimenet

A `standard kimenetre` egyetlen sort kell írni, ami Hanga kedvenc szavát tartalmazza.

Példa

Bemenet	Kimenet

`5 6`
`ha....`
`......`
`.n.g..`
`.....a`
`......`
	`hanga`

Bemenet	Kimenet

`3 3`
`ki.`
`.g.`
`.yo`
	`kigyo`

Korlátok

$1 \leq N, M \leq 6$

Időlimit: 1.0 s

Memórialimit: 256 MB

Pontozás

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	`N = 1`	19
2	a kedvenc szó betűi egymással szomszédos cellákban vannak	42
3	nincsenek további megkötések	39

Kormányablak

A kormányablakokban mostanság hatalmasra nőttek a sorok és ezzel a várakozási idők. Ezért a kormány úgy döntött, hogy az érkezéskor kapott sorszámok mostantól nem járnak le a nap végén, hanem a következő napokon is érvényesek maradnak. Hogy enyhítsék a helyzetet, közzétették a következő `N` napra, hogy az egyes napokon hány várakozót tudnak kiszolgálni, így elvileg mindenki meg tudja határozni, hogy melyik napon fog sorra kerülni.

Sajnos a legtöbb ismerősöd nem túl jó matekból, ezért a segítségedet kérik a megfelelő nap kiszámításához. Írj programot, ami `Q` ismerősöd számára megadja, hogy ki melyik napon fog sorra kerülni.

Bemenet

A `standard bemenet` első sorában a napok `N` száma és az ismerőseid `Q` száma szerepel.

A második sor `N` darab A_i egész számot tartalmaz: a kormányablak az `i`-edik napon A_i darab várakozót tud kiszolgálni.

A harmadik sor `Q` darab egész számot tartalmaz, a `j`-edik szám a `j`-edik ismerősöd **előtt** várakozók S_j száma.

Kimenet

A `standard kimenetre` összesen `Q` sort kell kiírni. A `j`-edik sor egyetlen pozitív egész számot tartalmazzon, annak a napnak a sorszámát, amikor a `j`-edik ismerősöd sorra kerül. Ha annyira sokan várakoznak, hogy a `j`-edik ismerősöd nem kerül sorra a következő `N` napban, akkor a `j`-edik sorba `-1`-et írj ki.

Példa

Bemenet	Kimenet
3 3 1 3 5 0 1 5	1 2 3

Az első példában:

- Az első ismerősöd előtt `0` várakozó van, így az első napon sorra kerül. Más már nem kerül sorra az első napon, mert $A_1 = 1$.
- A második ismerősöd előtt `1` várakozó van (az első ismerősöd), aki az első nap már sorra került. Így ő a második napon kerül sorra.
- A második nap végéig összesen `4` várakozó került sorra, így a harmadik ismerősöd (és a közvetlenül előtte sorban álló személy) csak a harmadik napon kerül sorra.

Bemenet	Kimenet
1 1 3 10	-1

Bemenet	Kimenet
5 6 0 2 3 4 0 7 3 2 10 5 9	4 3 3 -1 4 -1

Korlátok

$1 \le N \le 100\,000$

$1 \le Q \le 100\,000$

$0 \le A_i \le 15$ minden $i = 1 \ldots N$-re

$0 \le S_j \le 1\,000\,000$ minden $j = 1 \ldots Q$-ra

Időlimit: 1.5 s

Memórialimit: 256 MB

Pontozás

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	$Q = 1$ és $N \le 100$	50
2	nincsenek további megkötések	50

Online ételrendelés

Néhány barátoddal társasjátékos estét szerveztek, amihez az internetről szeretnétek rendelni a vacsorát. Összesen `N` résztvevő van, akiket `1`-től `N`-ig számozunk. A weboldalon `M` különböző étel közül lehet választani, amiket az `1, 2, ..., M` sorszámokkal azonosítunk. Mindenki kiválasztott pontosan egy ételt, amit rendelni szeretne magának. A `i`-edik résztvevő által választott étel sorszáma A_i. Ismert továbbá minden étel ára is: a `j`-edik étel ára C_j forint.

Mivel tudod, hogy az ilyen alkalmakkor a résztvevők rendszerint meg szokták kínálni egymást az ételükből, ezért szeretnéd elérni, hogy legalább `K` féle különböző étel legyen a rendelésben. Ehhez bármelyik résztvevőt (akár többüket is) rá tudod beszélni, hogy másik ételt válasszon, mint amit szeretett volna. Azonban az új ételnek az eredeti ételnél drágábbnak kell lennie, és a különbözetet neked kell kifizetned.

Írj programot, ami megmondja, hogy el tudod-e érni, hogy legalább `K` különböző étel legyen a rendelésben, és ha igen, akkor ehhez minimum mennyi pénzt kell pluszban költened.

Bemenet

A `standard bemenet` első sorában a résztvevők `N` száma, az ételek `M` száma és a `K` értéke szerepel.

A második sor `N` darab egész számot tartalmaz, a résztvevők által választott ételek A_i sorszámait.

A harmadik sor `M` darab egész számot tartalmaz, az ételek C_j árait.

Kimenet

A `standard kimenetre` egyetlen sort kell írni egyetlen egész számmal:

- Ha elérhető, hogy legalább `K` különböző étel legyen a rendelésben, akkor a változtatások miatt pluszban kifizetendő legkisebb pénzösszeget kell kiírni.
- Ha nem érhető el a cél, akkor `−1`-et kell kiírni.

Példa

Bemenet	Kimenet
3 4 3 1 1 2 1 2 3 4	2

Az első példában $K = 3$. Optimális megoldás, ha a második résztvevőt rábeszéljük, hogy az 1-es helyett a 3-as ételt válassza. Ekkor hárman három különböző ételt választanak, és a pluszban kifizetendő összeg $C_3 - C_1 = 3 - 1 = 2$ forint.

Bemenet	Kimenet
2 2 2 1 1 10 10	-1

A második példában $K = 2$ és a két résztvevő ugyanazt az ételt választotta. Mivel mindkét étel ára `10` forint, ezért egyiküket sem lehet rábeszélni, hogy másik ételt válasszon, így a cél nem érhető el.

Bemenet	Kimenet
2 2 1 1 2 10 10	0

Bemenet	Kimenet

`8 8 8`
`1 2 3 4 1 2 3 4`
`1 2 3 4 1000000000 1000000000 1000000000 1000000000`
	`3999999990`

Korlátok

$1 \le N, M \le 200\,000$

$1 \le K \le N$

$1 \le A_i \le M$ minden $i = 1 \ldots N$-re

$1 \le C_j \le 10^9$ minden $j = 1 \ldots M$-re

Időlimit: 2.0 s

Memórialimit: 256 MB

Pontozás

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	mindenki ugyanazt az ételt választotta	10
2	$K = N$	20
3	$N, M \le 15$	16
4	$C_j = j$ minden $j = 1 \ldots M$-re	23
5	nincsenek további megkötések	31

Nehéz nyomozás

A rendőrség egy veszélyes bűnöző után nyomoz a városban. A városban N lehetséges helyszín van, amiket 1-től N-ig számozunk. Ezeket M darab kétirányú út köti össze. Minden út két különböző helyszínt köt össze, és ismerjük az egyes utak hosszát. Két helyszínt legfeljebb egy közvetlen út köt össze. Bármely helyszínről bármely más helyszínre el lehet jutni az utak segítségével.

A rendőrség tudja, hogy a bűnöző nemrég egy ismeretlen X helyszínről egy szintén ismeretlen Y helyszínre utazott. Sikerült beszélniük K szemtanúval, akik látták az A_1, A_2, ..., A_K helyszíneken a bűnözőt az útja során. Sajnos a helyszínek időbeli sorrendje nem ismert, és az is lehet, hogy több más helyszínen is áthaladt a bűnöző. Az X és Y helyszínek sem biztos, hogy szerepelnek a szemtanúk által említett helyszínek között.

A rendőrség feltételezi, hogy a bűnöző sietett, ezért **egy lehető legrövidebb** útvonalon utazott X-től Y-ig, amely érinti az összes szemtanú által említett helyszínt. Írj programot, ami megmondja az összes lehetséges Y helyszínt. Az is lehet, hogy a rendőrség téved, és nem létezik a feltételeknek megfelelő Y.

Bemenet

A standard bemenet első sorában a helyszínek N száma, az utak M száma és a szemtanúk K száma szerepel.

A második sor K darab egész számot tartalmaz, a szemtanúk által említett A_i helyszíneket.

A következő M sor mindegyike három egész számot tartalmaz: U_j, V_j és L_j, ami azt jelenti, hogy van egy kétirányú út U_j és V_j helyszínek között, amelynek hossza L_j.

Kimenet

A standard kimenet első sorába a lehetséges Y helyszínek száma kerüljön - ha nincs ilyen, akkor 0. A második sorba a lehetséges Y helyszínek kerüljenek **növekvő sorrendben**.

Példa

Bemenet	Kimenet
6 6 2 1 5 1 2 1 2 3 1 3 4 1 4 5 1 5 6 1 6 1 1	4 1 2 4 5

Az első példában az 1, 2, 4, 5 helyszínek a lehetséges Y célpontok.
Például a 2-es helyszínre az $5 \to 6 \to 1 \to 2$ útvonalon érkezhetett.

Bemenet	Kimenet
4 3 3 2 3 4 1 2 3 1 3 5 1 4 4	0

A második példában nincs olyan legrövidebb útvonal két helyszín között, amely mindhárom megadott helyszínt érintené.

Bemenet	Kimenet

`7 9 2`
`5 1`
`2 3 3`
`2 1 1`
`3 1 2`
`3 5 2`
`1 4 1`
`5 4 3`
`5 6 5`
`5 7 2`
`6 7 1`
	`5`
	`1 2 5 6 7`

Bemenet	Kimenet

`5 4 2`
`2 4`
`2 5 1`
`2 1 1`
`1 4 1`
`4 3 1`
	`4`
	`2 3 4 5`

Korlátok

$2 \le N \le 200\,000$

$1 \le M \le 200\,000$

$1 \le K \le N$

$1 \le A_i \le N$ minden $i = 1 \ldots K$-ra

$1 \le U_j \ne V_j \le N$ minden $j = 1 \ldots M$-re

$1 \le L_j \le 10^9$ minden $j = 1 \ldots M$-re

Időlimit: 2.0 s

Memórialimit: 256 MB

Pontozás

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	$M = N - 1$ és a j-edik út a j és $j + 1$ helyszíneket köti össze	5
2	$M = N - 1$, vagyis bármely két helyszín között pontosan egy útvonal van	16
3	$N, M \leq 100$	15
4	$N, M \leq 1000$	17
5	$N, M \leq 100\,000$ és $K \leq 5$	15
6	nincsenek további megkötések	32