

Na řešení úloh máte 4,5 hodiny čistého času. Při soutěži je zakázáno používat jakékoliv pomůcky kromě psacích potřeb a přiděleného počítače (tzn. knihy, kalkulačky, mobily, apod.).

Řešením každého příkladu je zdrojový kód programu zapsaný v programovacím jazyce Pascal, C nebo C++. Řešení odevzdáváte pomocí soutěžního systému CMS, který ho automaticky otestuje na připravených sadách testovacích dat. Za každou úlohu můžete získat až 10 bodů. Detaily hodnocení naleznete v letáku s popisem prostředí. Podrobnější informace o testovacích datech najdete na konci zadání každé úlohy.

P-III-4 Vašek a houba

Program: houba.pas / houba.c / houba.cpp
Vstup a výstup: standardní vstup a výstup
Časový limit: najdete v soutěžním systému
Paměťový limit: najdete v soutěžním systému

Na běžkařském výletě potkal Vašek v lese kouzelnou houbu. „Já jsem kouzelná houba,“ povídá houba, „a splním ti jedno přání. Ale nejdřív musíš vyřešit následující hádanku.“

Soutěžní úloha

Na vstupu dostanete n různých přirozených čísel a_1, a_2, \dots, a_n . Vaším úkolem je vybrat několik z nich tak, aby jejich bitový AND byl nenulový a dělitelný co nejvyšší mocninou dvojky.

Nechť a, b jsou přirozená čísla. Potom c nazveme jejich bitovým ANDEM (a značíme $c = a \& b$), pokud je na i -té pozici dvojkového zápisu c jednička právě tehdy, když je jednička na i -té pozici dvojkového zápisu a i b . Pokud je některé z čísel a, b příliš malé, doplníme ho zleva nulami.

Pokud máme více čísel (a_1, a_2, \dots, a_k) , tak jako jejich bitový AND označíme $c = a_1 \& a_2 \& \dots \& a_k := (((\dots (a_1 \& a_2) \& a_3) \& \dots) \& a_k$. Tedy na i -té pozici c je jednička právě tehdy, když je na i -té pozici všech čísel a_1, a_2, \dots, a_k .

V C a C++ je operátor bitového ANDu značený jako $\&$, v Pascalu jako `and`.

Formát vstupu

Na prvním řádku dostanete jedno přirozené číslo, n . Na druhém řádku bude n různých mezerou oddělených přirozených čísel a_1, a_2, \dots, a_n .

Formát výstupu

Na první řádek vypište přirozené číslo $1 \leq m \leq n$, na druhý pak m mezerou oddělených různých přirozených čísel b_1, b_2, \dots, b_m takových, že $b_1 \& b_2 \& \dots \& b_m$ je

nenulové, dělitelné co největší mocninou dvojky a každé z nich je rovno některému ze vstupních čísel.

Příklady

Vstup:

5
6 5 7 1 2

Výstup:

3
7 6 5

$7 = (111)_2$, $6 = (110)_2$ a $5 = (101)_2$, z toho $7 \& 6 \& 5 = (100)_2 = 4$.

Vstup:

4
33 47 30 29

Výstup:

2
30 29

$33 = (100001)_2$, $47 = (101111)_2$, $30 = (11110)_2$ a $29 = (11101)_2$, z toho $30 \& 29 = (11100)_2 = 28$. Nejvyšší mocnina 2 dělící toto číslo je 4. Správnou odpovědí by také byla trojice 47, 29, 30, jejíž AND je 12.

Vstup:

4
536870913 268435458 134217732 8

Výstup:

1
8

Bodování

Plných 10 bodů dostanete za řešení, které úlohu vyřeší pro $1 \leq n \leq 10^5$ a $1 \leq a_i \leq 10^9$. V části vstupů (celkem za 3 body) bude platit $n \leq 20$. V části vstupů (celkem za 3 body) bude platit $a_i \leq 1024$. Ve vstupech za celkem 5 bodů bude platit alespoň jedna z podmínek $n \leq 20$ a $a_i \leq 1024$.

P-III-5 Poslední dostih sezóny

Program: dostih.pas / dostih.c / dostih.cpp
Vstup a výstup: standardní vstup a výstup
Časový limit: najdete v soutěžním systému
Paměťový limit: najdete v soutěžním systému

„Já si vsadím, že v posledním dostihu sezóny Šemík porazí Juráška.“

„Ale co to povidáš! Vždyť v minulém dostihu Jurášek porazil Pegase, před několika měsíci zase Pegas porazil Bleska, týden nato Blesk porazil Trojana a určitě si pamatuješ, jak onehdy Trojan doslova roznesl Šemíka na kopytech. Je přece očividné, že Šemík proti Juráškoví nemá šanci.“

„Tak to mi potom vysvětlí, proč Jurášek ve všech dostizích kromě toho minulého doběhl poslední...“

Soutěžní úloha

Letošního poháru v dostizích se účastní N koní očíslovaných 1 až N . Zatím se běželo K dostihů a vy máte k dispozici jejich výsledky. Kůň a si *věří* na jiného koně b , jestliže existuje posloupnost koní x_1, x_2, \dots, x_m taková, že a v nějakém závodě porazil x_1 , x_1 v nějakém závodě porazil x_2, \dots, x_m v nějakém závodě porazil b . Tato posloupnost může být i prázdná, tj. porazil-li kůň a koně b v nějakém závodě, pak si na něj věří. Pro každého koně určete, na kolik protivníků si věří.

Formát vstupu

Program čte vstupní data ze standardního vstupu. První řádek obsahuje přirozená čísla N a K ($N \cdot K \leq 10^6$). Na každém z následujících K řádků dostanete permutaci čísel 1 až N (výsledky dostihů, kůň na první pozici skončil první v příslušném závodě).

Formát výstupu

Program vypíše na standardní výstup jediný řádek, na němž bude N čísel, i -té z nich bude znamenat, na kolik koní si věří kůň i .

Příklady

Vstup:

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

Výstup:

4 4 4 4 4

Kůň číslo 1 porazil ve druhém dostihu všechny své čtyři soupeře, a proto si na všechny věří. V prvním dostihu jej ale všichni soupeři porazili, tedy si všichni věří na všechny.

Vstup:

4 3
1 3 4 2
3 4 1 2
4 1 3 2

Výstup:

3 0 3 3

Kůň číslo 2 skončil ve všech dostizích poslední a na nikoho si nevěří. Každý z ostatních koní vyhrál jeden z dostihů, a proto si věří na všechny ostatní.

Bodování

Plných 10 bodů dostanete za řešení, které úlohu vyřeší pro $N \cdot K \leq 10^6$ a $N \leq 500\,000$. V části vstupů za 7 bodů bude platit $N \cdot K \leq 10^6$ a $N \leq 5\,000$. V části vstupů za 5 bodů bude platit $N \cdot K \leq 10\,000$ a $N \leq 500$. V části vstupů za 3 body bude platit $N \leq 10$ a $K \leq 2$.

P-III-6 Vykopávky

Program: vykopavky.pas / vykopavky.c / vykopavky.cpp
Vstup a výstup: standardní vstup a výstup
Časový limit: najdete v soutěžním systému
Paměťový limit: najdete v soutěžním systému

Slavný archeolog Indiana Jones přijel do Kocourkova studovat jeho bohatou historii. Zejména se zajímá o Kocourkovské podzemí, jehož mapu se mu podařilo nalézt v archívech.

Podzemí má obdélníkový tvar skládající se z $R \times C$ jednotkových čtverců. V některých z těchto čtverců jsou podzemní prostory zabírající celou jejich plochu. Všechny tyto podzemní prostory jsou ve stejné hloubce, takovému prázdné čtverce propojené přes hrany (ale nestačí diagonálně) tedy tvoří větší souvislé místnosti.

Profesor Jones by rád prozkoumal co nejvíce takových souvislých místností (chce maximalizovat jejich počet, nikoliv plochu). Kocourkovští radní mu ale dovolili vykopat pouze jedinou díru obdélníkového tvaru a předepsaných rozměrů $M \times N$. Z této díry může navštívit všechny místnosti, které ji protínají nebo s ní alespoň sdílí hranu. Pomozte mu nalézt nejvhodnější polohu vykopané díry.

Formát vstupu

První řádek vstupu obsahuje čtyři přirozená čísla R , C , M a N oddělená mezerami; čísla R a C popisují rozměry mapy ($1 \leq R, C \leq 800$) a čísla M a N určují rozměry vykopané díry ($1 \leq M \leq R, 1 \leq N \leq C$).

Na dalších R řádcích je zadána mapa podzemí. Každý z nich obsahuje C znaků popisujících čtverce mapy: '.' označuje čtverec obsahující podzemní prostory a '#' označuje plný čtverec.

Formát výstupu

Vypište jedno celé číslo: největší počet místností, které profesor Jones může prozkoumat.

Příklady

<i>Vstup:</i>	<i>Výstup:</i>
3 4 1 2	2
..##	
.##.	
##..	

Na zadané mapě jsou dvě místnosti, které lze prozkoumat po vykopání dvou prostředních čtverců.

<i>Vstup:</i>	<i>Výstup:</i>
2 2 2 2	0
##	
##	

Na této mapě nejsou žádné místnosti.

Vstup:

3 3 1 1

.##

###

##.

Výstup:

1

Vykopáním jednoho čtverce nelze dvě místnosti z této mapy propojit, jelikož diagonální dotyk nestačí.

Bodování

Plných 10 bodů dostanete za řešení, které úlohu vyřeší pro $R, C \leq 800$. V části vstupů za 6 bodů bude platit $R, C \leq 350$. V části vstupů za 3 body bude platit $R, C \leq 100$ a $N, M \leq 30$.