

Ліфтом чи пішки

Input file name:	lift.in
Output file name:	lift.out
Time limit:	100 ms
Memory limit:	256 M

Степан живе на N -му поверсі. Коли Степан заходить в під'їзд, він дивиться, на якому поверсі в цей момент знаходиться ліфт і вирішує, викликати ліфт або піти по сходах.

Сьогодні ліфт знаходиться на K -му поверсі. Степан заходить в під'їзд на 1 поверсі. Він піднімається на один поверх за A секунд. Ліфт переміщається на один поверх за B секунд. Часом входу в ліфт і виходу з ліфта, а також переміщення до сходів і назад можна знехтувати.

Допоможіть Степану прийняти рішення, виведіть, за який час він потрапить на свій поверх на ліфті і по сходах, відповідно.

Вхідні дані: У єдиному рядку вхідного файлу знаходяться чотири цілих числа N, K, A, B ($2 \leq N \leq 100, 1 \leq K \leq 100, 1 \leq A, B \leq 10^3$).

Вихідні дані: Виведіть два цілих числа: час, за який Степан підніметься на свій поверх на ліфті, і час, за який Степан підніметься на свій поверх по сходах.

Пояснення до прикладу: У прикладі ліфту необхідна $7 \times 3 = 21$ секунда, щоб спуститися з 8 поверху і потім $14 \times 3 = 42$ секунди, щоб піднятися на 15 поверх, де живе Степан. Степану ж необхідно $14 \times 5 = 70$ секунд, щоб піднятися на 15 поверх сходами.

Приклад вхідних та вихідних даних:

lift.in	lift.out
15 8 5 3	63 70

Англійська від Степана

Input file name:	english.in
Output file name:	english.out
Time limit:	100 ms
Memory limit:	256 M

Степан вивчає стародавню англійську мову. Оскільки він робить це в рамках домашнього завдання на завтра, він не встигає детально вивчити його граматику, тому він вирішив для простоти застосувати наступні правила для перекладу сучасних англійських слів у свій варіант стародавньої англійської.

- Усі літери «s», після яких не йде «h» і які не є першими в слові, замінюються на комбінацію «th».

- Якщо перша буква в слові «e», то вона замінюється на «ae».

- Комбінація «oo» замінюється на «ou», причому якщо в слові йде поспіль більше двох букв «o», то з них замінюються тільки перші дві.

Допоможіть Степану перевести кілька слів на свою версію стародавньої англійської мови.

Вхідні дані: Перший рядок введення містить число N ($1 \leq N \leq 100$) - кількість слів, які потрібно перевести. Далі йдуть N рядків, кожен з яких складається тільки з букв латинського алфавіту. Всі букви кожного слова маленькі, крім, можливо першої, яка може бути заглавною. Довжина кожного слова не перевищує 30.

Вихідні дані: Виведіть N рядків - результат перекладу. Якщо перша літера вихідного слова була заглавною, то такою ж має бути і перша буква перекладеного слова. Інакше всі букви повинні залишитися маленькими.

Приклад вхідних та вихідних даних:

english.in	english.out
3	soun
soon	Aenglish
English	thith
this	

Подарунки від Діда Мороза

Input file name:	gift.in
Output file name:	gift.out
Time limit:	200 ms
Memory limit:	256 M

Степан справді вірить в Діда Мороза. Він вибирає собі подарунки на новий рік. Степан знає, що Дід Мороз купить йому рівно два подарунки: один нібито від мами, а інший нібито від тата.

У магазині, де Дід Мороз буде купувати подарунки, продається N подарунків, про кожен подарунок відома його ціна: ціна i -го подарунка дорівнює A_i грн. Степан знає, що Дід Мороз може витратити на покупку його подарунків не більше X рублів. Зрозуміло, він хоче отримати якомога більш дорогі подарунки.

Таким чином, він хоче вибрати два різних подарунка з максимальною сумарною ціною, але при цьому вона не повинна перевищувати X .

Допоможіть Степану вибрати собі подарунки.

Вхідні дані: Перший рядок введення містить два цілих числа: N і X ($2 \leq N \leq 100000, 2 \leq X \leq 10^9$).

Другий рядок введення містить N цілих чисел: $A_1, A_2, \dots, A_N$ ($1 \leq A_i \leq 10^9$). Гарантується, що існує два подарунки з сумарною ціною не більше X .

Вихідні дані: Виведіть одне ціле число: максимальну сумарну ціну двох різних подарунків, що не перевищує X .

Приклад вхідних та вихідних даних:

gift.in	gift.out
5 13	9
2 1 5 4 3	

Гра зі стільцями

Input file name:	chairs.in
Output file name:	chairs.out

Time limit:	500 ms
Memory limit:	256 M

Вчитель з математики ЛІ є організатором розважальних конкурсів в Фіцеї. Фінальним конкурсом буде відома всім "Гра зі стільцями". Її правила наступні: є декілька стільців, їх кількість повинна бути на один менше, ніж кількість учасників. Наприклад, якщо гравців сім — стільців шість. Коли звучить музика, всі бігають навколо стільців. Як тільки музика зупиниться, гравці швидко повинні сісти на стільці. Кому місця не дісталось той вибуває з гри. Потім один стілець забирається і гра продовжується поки не залишиться один учасник.

ЛІ знає, що нагорода за фінальний конкурс дуже цінна — книжки з математики. Звичайно, вона хоче, щоб їх отримав її найулюбленіший учень, Степан. Як організатор, ЛІ може казати коли зупиняти музику, тому вона хоче визначити такий момент, щоб Степан точно встиг зайняти стілець.

Більш формально, поле для гри можна уявляти як неорієнтований граф з N вершин та M ребер, кожне ребро якого має певну довжину. В деяких вершинах графу знаходяться стільці, деякі вершини представляють з себе місце, в якому може знаходитись якийсь учасник під час гри музики, інші вершини виступають в ролі проміжних, через які може лежати шлях учасника до стільця.

ЛІ хоче знати найкоротший шлях від якоїсь вершини з учасником до якоїсь вершини зі стільцем. Таким чином, вона зупинить музику тоді, коли Степан буде знаходитись в цій вершині і він буде мати перевагу в тому, що найкоротший шлях до стільця в нього буде не більший, ніж в будь-якого з інших учасників.

Вхідні дані: В першому рядку знаходяться два числа N і M ($2 \leq N \leq 70000$, $1 \leq M \leq 200000$) — кількість вершин і ребер, відповідно. В другому рядку знаходиться N цілих чисел A_i ($0 \leq A_i \leq 2$). Якщо $A_i = 0$, то в цій вершині нема ні стільця, ні учасника. Якщо $A_i = 1$, то в цій вершині знаходиться учасник. Якщо $A_i = 2$, то в цій вершині знаходиться стілець. Гарантується, що хоча б одне з цих чисел дорівнює одиниці, і хоча б одне дорівнює двійці.

В кожному з наступних M рядків знаходиться по три числа X_i , Y_i , Z_i , які позначають, що між вершинами X_i і Y_i ($1 \leq X_i, Y_i \leq N$, $X_i \neq Y_i$, $0 \leq Z_i \leq 100000$) з'єднані ребром, довжина якого дорівнює Z_i хвилин. Гарантується, що два вершини з'єднані не більш ніж одним ребром.

Вихідні дані: Якщо існує такий шлях, по якому Степан може добратись з початкового місця до стільця якнайшвидше, то виведіть три числа: x , y , d , які позначають, що з вершини x є шлях до вершини y з сумарним часом d хвилин. Додатково мають виконуватись умови $A_x = 1$, $A_y = 2$.

Якщо такого шляху не існує виведіть -1.

Якщо існує декілька відповідей, виведіть будь-яку з них.

Приклад вхідних та вихідних даних:

chairs.in	chairs.out
6 6	1 6 5
1 1 0 0 2 2	
1 3 3	
1 2 3	
1 5 6	
3 6 2	
2 3 4	
2 4 5	

Новий алгоритм шифрування

Input file name:	encryption.in
Output file name:	encryption.out
Time limit:	300 ms
Memory limit:	128 M

Фірма, на яку влаштовується Степан, займається розробкою нового наднадійного алгоритму шифрування RSA-SUPER. Степан дізнався, що RSA (аббревіатура від прізвищ Rivest, Shamir і Adelman) - криптографічний алгоритм з відкритим ключем. Криптосистема RSA стала першою системою, придатною і для шифрування, і для цифрового підпису. Алгоритм використовується у великому числі криптографічних додатків, включаючи PGP, S / MIME, TLS / SSL, IPSEC / IKE та інших.

Як відомо, в основі алгоритму RSA лежить використання пари простих натуральних чисел P і Q і утвореного числа $N = P * Q$. Числа P і Q називаються ключами шифрування, а число N - модулем шифрування. Просте число - це натуральне число, яке має рівно два різних натуральних дільники: одиницю і самого себе.

Принциповою відмінністю нового RSA-SUPER алгоритму від RSA алгоритму полягає у виборі ключів. Якщо в реалізації RSA алгоритму потрібна пара простих чисел P і Q , то в RSA-SUPER алгоритмі числа P і Q повинні бути взаємно простими. Два натуральних числа називаються взаємно простими, якщо вони не мають жодних спільних дільників, крім одиниці.

Для аналізу надійності нового алгоритму власники фірми хочуть дізнатися кількість різних пар ключів P і Q , таких, що $1 < P < Q$ і відповідний їм модуль шифрування задовольняє умові: $N \leq K$. Саме це нелегке завдання і було дано Степану, а він, звісно просить допомоги у вас.

Вхідні дані: Перший рядок вхідного файлу містить одне ціле число K ($1 \leq K \leq 10^9$).

Вихідні дані: Вихідний файл повинен містити одне ціле число - кількість різних пар ключів P і Q .

Оцінювання:

$K \leq 300$ - не менше 20 балів

$K \leq 1000$ - не менше 30 балів

$N \leq 5000$ - не менше 40 балів

$N \leq 10^5$ - не менше 50 балів

$N \leq 10^6$ - не менше 60 балів

Пояснення: У першому прикладі: (2,3); (2,5); (3,4)

У другому прикладі: (2,3); (2,5); (2,7); (2,9); (3,4); (3,5)

Приклад вхідних та вихідних даних:

encryption.in	encryption.out
12	3
18	6

Input file name:	illuminated.in
Output file name:	illuminated.out
Time limit:	100 ms
Memory limit:	256 M

Степан з метою додаткового заробітку вирішив встановити в кожному містечку надсучасний ліхтар, промені якого паралельні поверхні землі, щоб освітлити вулицю. На вулиці розташовані будівлі, задані висотами стін в метрах. Будівля вважається освітленою, якщо для прямих променів ліхтаря відкритий хоча б один метр стіни. Степан вирішив підрахувати кількість освітлених будівель, щоб знати скільки він зможе заробити. Допоможіть Степану - напишіть програму, яка вирішить його проблему.

Вхідні дані: Перший рядок вхідного файлу містить ціле число N ($1 \leq N \leq 100000$) - кількість будівель в кожному містечку. У другому рядку записано від N чисел - висоти будівель H_i ($1 \leq H_i \leq 100000$).

Будемо вважати, що всі будівлі розташовуються в один ряд на вулиці, орієнтованій зі сходу на захід, ліхтар освітлює їх з боку першої будівлі.

Вихідні дані: Єдиний рядок вихідного файлу має містити одне ціле число - кількість освітлених будівель.

Малюнки до прикладів:



Приклади вхідних та вихідних даних:

illuminated.in	illuminated.out
4	3
1 2 2 3	
5	2
2 1 3 1 2	

Завдання з біології

Степан, готуючись до олімпіади з програмування, пропустив багато уроків біології, і щоб виправити становище, взявся за непросте завдання. Це завдання полягає в аналізі зображень, отриманих за допомогою електронних мікроскопів.

Перед проведенням аналізу зображення проходять кілька стадій фільтрації. На першій стадії зображення перекладається в монохромне, тобто що має тільки два кольори: білий та чорний. Друга стадія є більш складною і полягає у видаленні шумів із зображення шляхом підбору фільтра. Але це виявилось непростим, оскільки для різних видів зображень потрібні різні фільтри. З метою підбору відповідного фільтра було вирішено перед другою стадією фільтрації проводити додаткові розрахунки.

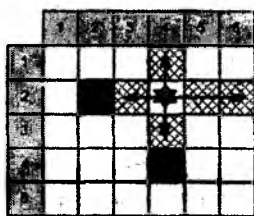
Додаткові розрахунки полягають у знаходженні коефіцієнта зв'язності вихідного зображення.

Коефіцієнт зв'язності - це число різних пар сусідніх пікселів.

Зображення являє собою прямокутну таблицю пікселів розміру $N \times M$.

Кожен піксель має або білий колір, або чорний. Два різних пікселя називаються сусідніми, якщо виконуються наступні умови:

- Обидва пікселя мають білий колір.
- Обидва пікселя знаходяться або в одному і тому ж стовпці або в одному і тому ж рядку.
- Всі пікселі, що перебувають або в рядку, або в стовпці між зазначеними пікселями, є білими.



Малюнок №1: $N = 5$ і $M = 6$, приклад сусідніх символів для пікселя (2, 4).

Для наведеного вище прикладу всі зафарбовані пікселі є сусідніми для пікселя, позначеного зірочкою. Дві пари сусідніх пікселів, які можуть бути отримані одна з іншою за допомогою перестановки елементів, не вважаються різними. Тобто пара сусідніх пікселів $\{(2; 4), (2; 6)\}$ і пара $\{(2; 6), (2; 4)\}$ не рахуються різними.

Степану потрібно написати програму, яка по заданому опису зображення розраховує коефіцієнт зв'язності даного зображення. Допоможіть йому.

Вхідні дані: Перший рядок вхідного файлу містить два цілих числа N і M ($1 \leq N, M \leq 200$). Далі йдуть N рядків по M символів в кожному, що описують задане монохромне зображення. Символ '.' (ASCII 46) - позначає білий символ, а '#' (ASCII 35) - чорний.

Вихідні дані: Одне ціле число - коефіцієнт зв'язності вихідного монохромного зображення.

Приклади вхідних та вихідних даних:

biology.in	biology.out
2 3	5
...	
.#.	
3 4	13
...#	
.#..	
...#	

Гра з рядком

Степан дуже подобає паліндроми. Паліндромом називається рядок, який однаково читається як зліва направо, так і справа на ліво.

Підпаліндромом даного рядка називається послідовність символів з даного рядка, які не обов'язково йдуть посліпіль, і є паліндромом.

Наприклад, *ABA* є підпаліндромом рядка *ACBFADEF*.

Степан вирішив пограти з рядком. Він придумує рядок і бере з даного рядка підрядок (послідовність символів з l по r , що йдуть посліпіль) та визначає паліндром максимальної довжини. Потім бере новий підрядок і вже з нього визначає паліндром максимальної довжини, і так k разів.

Напишіть програму, яка допоможе Степану.

Вхідні дані: Перший рядок вхідного файлу містить рядок довжиною не більше 100 символів з великих букв латинського алфавіту.

Другий рядок містить одне число k ($1 \leq k \leq 10$) - кількість разів, яку Степан зіграв з даним рядком.

Наступні k рядків містять по два числа l та r - номери першого та номер останнього символа підрядка.

Input file name:	string.in
Output file name:	string.out
Time limit:	300 ms
Memory limit:	256 M

Вихідні дані: Вихідний файл має містити k рядків, у кожному з яких має бути записана довжина максимального підпаліндрому та через пробіл сам максимальний підпаліндром для відповідних l та r у вхідних даних. Якщо таких підпаліндромів кілька, то виведіть любий з них.

Приклади вхідних та вихідних даних:

string.in	string.out
ACBFADEF	3 ABA
3	1 C
1 5	3 FAF
2 6	
2 8	

Адміністративна реформа

В Ужляндії вирішили провести адміністративно-територіальну реформу. Як відомо, в Ужляндії є N міст, розташованих уздовж довгої прямої дороги, i -е місто розташоване на відстані X_i ($0 \leq X_1 < X_2 < \dots < X_N \leq 10^9$) кілометрів від початку дороги.

В рамках реформи було прийнято рішення розділити Ужляндію на K регіонів. Кожне місто має увійти рівно в один регіон.

У кожен регіон увійде від A до B міст, до того ж, ці міста повинні мати номери, що йдуть поспіль. Таким чином, кожен регіон характеризується числами i та l , для яких $l \leq i, i + l - 1 \leq N, A \leq l \leq B$ і в регіон входять міста з номерами $i, i + 1, \dots, i + l - 1$.

Щоб мінімізувати витрати на обслуговування регіонів, адміністрація хоче, щоб максимальна відстань між містами, що входять в одну провінцію, було якомога менше.

Допоможіть виконати поділ Ужляндії.

Вхідні дані: Перший рядок вхідного файлу містить чотири цілих числа: N, K, A, B ($1 \leq N \leq 200, 1 \leq K \leq N, 1 \leq A \leq B \leq N, AK \leq N \leq BK$). Другий рядок вхідного файлу містить N цілих чисел: $X_1, X_2, \dots, X_N$ ($0 \leq X_1 < X_2 < \dots < X_N \leq 10^9$).

Вихідні дані: Виведіть одне ціле число: мінімально можливе Z , таке щоб можна було розбити міста на регіони описаним чином, і відстань між містами всередині одного регіону не перевищувала Z .

Пояснення до прикладу: У прикладі оптимально перші 4 міста об'єднати в перший регіон, а п'яте і шосте - у другий. Максимальна відстань між двома містами в одному регіоні: $13 - 6 = 7$.

Приклади вхідних та вихідних даних:

reform.in	reform.out
6 2 2 4	7
1 2 3 4 6 13	

Степан і "Великі любителі Гаррі Поттера"

Степан дуже любить серію фантастичних романів "Гаррі Поттер". Він так сильно нею захоплюється, що входить в групу "Великі любителі Гаррі Поттера" та навіть склав збірку всіх заклинань, яким вчать в Гогвартс. Також Степан вивчив всі правила та закономірності цих заклинань.

Самі по собі заклинання нічого не варті, їх потрібно вимовляти в певній послідовності, яка називається комбінацією заклинань.

Ця послідовність має починатись з базового заклинання, та закінчуватись на заклинання, яке називається основним.

Поряд в комбінації можуть йти лише, так звані, спарені заклинання, наприклад, після базового можна промовити лише спарені з ним, Степан визначив які заклинання є спареними. Комбінація записується як конкатенація заклинань (без проміжків між ними). Степан визначив яким має бути основне (останнє) заклинання в комбінації, тепер він хоче представити свою комбінацію публіці. Для того, щоб група любителів Гаррі Поттера прийняла цю комбінацію, необхідно, щоб її довжина була найменшою серед всіх можливих (адже так простіше буде вчити). Також Степан помітив, що лексикографічно малі комбінації краще приймаються публікою, хто знає, можливо це важливо. Тому необхідно серед всіх можливих найкоротших комбінацій вибрати лексикографічно найменшу. Допоможіть Степану знайти таку комбінацію.

Вхідні дані: В першому рядку знаходиться два цілих числа N і M ($2 \leq N \leq 1000, 1 \leq M \leq 250000$) — кількість заклинань та кількість пар спарених заклинань відповідно. Далі в N рядках записані самі заклинання, до того ж, перше заклинання — базове, останнє заклинання — заклинання на яке комбінація має закінчуватись. Всі заклинання складаються лише з маленьких літер латинського алфавіту, їх довжини не перевищують 1000 символів. Далі в M рядках записано по два різних числа A_i і B_i — номери заклинань, які є спареними. Пари заклинань не повторюються.

Вихідні дані: В єдиному рядку виведіть шукану комбінацію.

Оцінювання:

$N \leq 50$, назви заклинань з номерами від 1 до N складаються з однієї букви — не менше 30 балів

$N \leq 50$ — не менше 60 балів

$N \leq 250$ — не менше 75 балів

Приклади вхідних та вихідних даних:

spell.in	spell.out
2 1	avada
avada	
kedavra	
1 2	
4 4	sectumcruciosempra
sectum	
crucio	
reparo	
sempra	
1 2	
1 3	
2 4	
3 4	