Лабораторна робота №3

## Завдання 1

Перестановкою P[1..n] розміру n називається набір чисел від 1 до n, розташованих у порядку. При цьому в ньому має бути рівно один раз кожне з цих чисел. Прикладом перестановок є 1,3,4,5,2 (n=5) і 3,2,1 (n=3), а, наприклад, 1,2,3,4,5,1 перестановкою не є, оскільки число 1 зустрічається двічі.

Число i називається нерухомою точкою для перестановки P, якщо P[i] = i. Наприклад, у перестановці 1,3,4,2,5 рівно дві нерухомі точки: 1 і 5, а перестановка 4,3,2,1 не має нерухомих точок.

Дано два числа: n і k. Знайдіть кількість перестановок розміру n із рівно k нерухомими точками.

## Вхідні дані

Вхідний файл INPUT.TXT містить два цілих числа n (1 ≤ n ≤ 9) і k (0 ≤ k ≤ n).

## Вихідні дані

У вихіднийфайл OUTPUT.TXT виведіть відповідь на завдання.

## Приклади

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 5 2 | 20 |
| 2 | 9 6 | 168 |
| 3 | 2 1 | 0 |
| 4 | 9 0 | 133496 |

**Завдання 2**

Великий концертний зал Байтланда - відоме на весь світ місце, де мріють виступати найбільші оперні співаки та співачки. У бажаючих послухати їхні виступи, як правило, не бракує. Тому Міністерство культури Байтланда вирішило збільшити розміри зали.

Однак це викликало іншу проблему: на задніх лавах оновленої концертної зали відвідувачі практично нічого не чують. Тому Інженерний інститут запропонував проект акустичної системи, яка складатиметься з мікрофонів, що записують те, що відбувається на сцені, та динаміків, що транслюють посилений звук у зал.

В ідеальному випадку динаміки повинні достовірно відтворювати звук, який записується з мікрофонів. На практиці цього досягти майже неможливо, тому що при поточному рівні розвитку технологій практично всі динаміки відтворюють різні частоти з різною гучністю. Миритися з цим інженери, однак, не збираються.

У лабораторії вдалося виміряти АЧХ (амплітудно-частотну характеристику) динаміків та подати її у наступній формі:

1. весь діапазон частот, що відтворюються динаміками, розділений на N послідовних інтервалів, що нумеруються від 1 до N;
2. в i-му інтервалі відомо Ai - значення усередненої по інтервалу гучності вдецибелах(Прим.: децибел(дБ)-одиницявимірювання гучності).

*АЧХ з прикладу №2 та підсилювачі, які потрібні, щоб її «вирівняти» на рівні*

*4 дБ (5 штук, підсилювачі застосовуються в порядку від верхніх до нижніх)*

АЧХ динаміків можна правити за допомогою спеціальних електронних пристроїв – підсилювачів. Підсилювач може підняти силу сигналу на всіх інтервалах з A-го по B-й на 1 дБ, де A, B - довільні натуральні числа, що не перевищують N, A ≤ B. Через особливості застосовуваних у пристрої радіодеталей, сила сигналу перед застосуванням підсилювача повинна бути однакова на всіх інтервалах з A по B-ой. Так як вартість підсилювача досить велика, то їхня кількість повинна бути мінімально.

Вважається, що динаміки відтворюють звук достовірно, якщо всіх інтервалах значення гучності однаково. Напишіть програму, яка обчислить мінімальну кількість підсилювачів, потрібну для досягнення достовірного звучання.

# Вхідні дані

Перший рядок вхідного файлу INPUT.TXT містить одне число N (1 ≤ N ≤ 200000) – кількість інтервалів, на яких замірялася АЧХ. Другий рядок містить N натуральних чисел Ai (1 ≤ i ≤ N, 1 ≤ Ai ≤ 109), розділених одиночними пробілами - усереднена гучність на i-му інтервалі в децибелах.

# Вихідні дані

Єдиний рядок вихідного файлу OUTPUT.TXT повинен містити одне ціле число - мінімальне число підсилювачів, за допомогою яких можна досягти достовірного звучання.

# Приклади

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 31 3 2 | 3 |
| 2 | 41 2 4 2 | 5 |
| 3 | 53 1 4 1 1 | 6 |

## Завдання 3

Десь далеко від нас, на краю землі, є одна невелика, але гарна країна WWW з найбагатшим історичним минулим. Люди, які її населяють, відомі всьому світу своєю добротою і гостинністю. Уся територія країни умовно поділена на райони. Кожен район складається із певної кількості міст, одне з яких є районним центром.

Деякі пари міст цієї держави з'єднані двосторонніми дорогами, причому відомо, що будь-який районний центр з'єднаний дорогами з рештою міст району, а також не більше ніж з двома іншими районними центрами. Жодна дорога не з'єднує два міста, які не є районними центрами. Дорога може з'єднувати два міста з різних районів тільки в тому випадку, якщо вони є районними центрами. Між будь-якою парою міст може бути не більше однієї дороги. Дороги побудовані таким чином, що ними можна доїхати з будь-якого міста в будь-який інший.

Історично склалося, що право на володіння всіма дорогами донедавна належало одній відомій компанії «АвтоДор». У зв'язку з цим до конституційного суду було подано антимонопольний позов, який був задоволений – тепер компанії належить продати частину своїх володінь. Економісти компанії визначили для кожного шляху її вартість.

Одна маленька, небагата, але горда фірма "КурсІнвест", в якій Ви працюєте фінансовим директором, захотіла викупити частину доріг, а саме, K з них. Причому необхідно, щоб для будь-яких двох міст, до яких примикає хоча б одна з викуплених K доріг, існувало не менше одного шляху, що з'єднує їх, що складається тільки з придбаних доріг. Вам як фінансовому директору було доручено знайти економічно вигідне рішення. Рішення називатимемо економічно вигідним, якщо грошова сума, витрачена на придбання доріг, є мінімальною.

# Вхідні дані

У першому рядку вхідного файлу INPUT.TXT знаходяться три цілих числа N (3 ≤ N ≤ 2000), M та К (1 ≤ K < M ≤105), де N – загальна кількість міст, M – загальна кількість доріг, K – кількість доріг, що необхідно придбати.

Далі слідує M рядків, у кожному з яких записані три цілих числа Ai, Bi та Ci, де Ai та Bi

– номери міст, які з'єднані дорогою (1 ≤ Ai, Bi ≤ N, Ai ≠ Bi), а Ci – вартість дороги (1 ≤ Ci ≤ 106). Усі числа у рядках розділені одиночними пробілами.

# Вихідні дані

Вихідний файл OUTPUT.TXT повинен складатися з рядків K, кожен з яких повинен містити одне число – номер придбаної дороги. Дороги нумеруються в порядку їхнього введення починаючи з одиниці. Якщо рішень кілька, виведіть будь-яке з них.

# приклад

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 984 892 5110 3811 257 878 5612 479 3 5 5 | 4183 |

## Завдання 4

У день народження Вови батьки вирішили запросити всіх його друзів. Після того, як хлопці випили сік та з'їли яблучний пиріг, вони вирішили пограти у захоплюючу гру Green Darts, правила якої досить прості:

* У грі беруть участь N людей.
* Кожен гравець 3 рази поспіль кидає дротик у ціль. Кожен кидок, залежно від точності, може принести гравцю цілу кількість очок від 0 до 100.

1. Після того, як всі кидки зроблені, кількість очок, набраних кожним гравцем позакінчення гри, обчислюється наступною формулі:Si = ai1 \* p2 + ai2 \* p + ai3,

де Si – підсумкова кількість очок, яку набрав перший гравець; aij - кількість очок, яке приніс i-го гравця зроблений ним j-ий кидок; p – однакове для всіх гравців дійсне число (0 < p ≤ 1).

* Потім для кожного гравця визначається місце, яке він зайняв після закінчення гри.Для цього використовується наступна формула:Mi= Ki+ 1,

де Mi - місце, яке посів перший гравець після закінчення гри; Ki – кількість гравців, які набрали більшу підсумкову кількість очок, ніж гравець. Неважко бачити, що кілька гравців могли розділити те саме місце після закінчення гри.

* Після цього формується результат гри. Підсумок гри – це набір із N цілих чисел; i-е число в цьому наборі дорівнює місцю, яке зайняв i-ий гравець після закінчення гри. Два набори з N чисел вважаються різними, якщо існує таке i, що i-е число в першому наборі не дорівнює i-му числу в другому наборі.

Вибір значення числа гості вирішили надати винуватцю урочистості. Вова відразу задумався над тим, як його вибір може вплинути на підсумок гри і багато залежить від вибору числа p. Вова замислився над відповіддю на таке запитання: скільки існує різних підсумків гри, залежно від вибору значення p? Допоможіть Вові розібратися із цим нелегким питанням.

# Вхідні дані

Перший рядок вхідного файлу INPUT.TXT містить ціле число N – кількість учасників гри (1 ≤ N ≤ 100). Кожен із наступних N рядків містить 3 цілих числа ai1, ai2, ai3, розділені одиночними пробілами, що визначають кількість очок, яку принесли i-му гравцеві перший, другий та третій кидки відповідно (0 ≤ aij ≤ 100). Усі числа цілі.

# Вихідні дані

Вихідний файл OUTPUT.TXT повинен містити ціле число, яке дорівнює кількості різних підсумків гри, залежно від вибору значення числа p.

# приклад

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 31991 100 100 10051 49 13 | 3 |

## Завдання 5

*Діаграми Юнга*використовуються для того, щоб зобразити розбиття числа на доданки. Розбиття числа n на складові є сумою виду n = m1 + m2 + … + mk, де m1 ≥ m2 ≥ … ≥ mk.

Діаграма складається з n квадратиків, організованих як k рядів, де k кількість доданків у розбиття. Ряд, що відповідає числу mi, містить mi квадратиків. Всі ряди вирівняні по лівому краю і впорядковані від довшого до більш короткого.

Наприклад, діаграма Юнга, наведена малюнку, відповідає розбиттю 10 = 5 + 3 + 2.

Іноді можна вписати одну діаграму Юнга до іншої. Діаграму X можна вписати в діаграму Y, якщо можна видалити деякі квадратики з діаграми Y так, щоб вийшла діаграма X. Зазначимо, що дозволяється лише видаляти деякі квадратики, обертати або відображати діаграму не дозволяється. Наприклад, діаграма для розбиття 5 = 3

+ 2 можна вписати в діаграму для розбиття 10 = 5 + 3 + 2, як показано на малюнку.

З іншого боку, діаграму для розбиття 8 = 4+4 не можна вписати в діаграму для розбиття 10 = 5 + 3 + 2. Для заданого n знайдіть таке розбиття n на доданки, що у відповідну діаграму Юнга можна вписати максимальну кількість різних діаграм. Наприклад, діаграму для розбиття 10 = 5 + 3 + 2 можна вписати 36 різних діаграм. Однак це не максимальне значення. У діаграму для розбиття 10 = 4+2+2+

1+1 можна вписати 41 діаграму Юнга.

# Вхідні дані

Вхідний файл INPUT.TXT містить ціле число n (1 ≤ n ≤ 100).

# Вихідні дані

На першому рядку вихідного файлу OUTPUT.TXT виведіть максимальну кількість діаграм Юнга, які можна вписати в деяку діаграму, що відповідає розбиттю на доданки n. На другому рядку виведіть одне чи більше цілих чисел - кількість квадратиків у кожному з рядів оптимальної діаграми.

# приклад

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 10 | 414 2 2 1 1 |

## Завдання 6

Нарешті у селах Віллорібо та Віллобаджо закінчилися свята. Перемито весь посуд! Цей процес пройшов так швидко і невимушено, що братами Карліоне було вирішено відкрити мережу агентств Clear World and Brothers, що спеціалізуються на професійному миття посуду. В області Нова Берляндія, де знаходяться знамениті села, всього N сіл. Система координат введена так, що Віллорібо має координати (x1, 0), а Віллобаджо - (x2, 0). Координати всіх сіл цілі числа не перевищують за модулем 106. Ви працюєте на містера Берліоні старшого і ваше завдання знайти оптимальне розташування для регіонального відділення «Clear World and Brothers»,

# Вхідні дані

У першому рядку вхідного файлу INPUT.TXT записано натуральне число N (2 ≤ N ≤ 15000). Далі у N рядках записані пари координат усіх вершин. Віллорібо та Віллобаджо перше і друге село відповідно. Можливо, що кілька сіл розташовані так близько, що їхні координати збігаються.

# Вихідні дані

У вихідний файл OUTPUT.TXT виведіть абсцис оптимального розташування агентства. Дозволяється абсолютна похибка трохи більше одиниці.

# приклад

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 4-100 100 31 1 -1 | 2.000000 |

## Завдання 7

Даний прямокутник на координатній площині з лівим нижнім кутом у точці (0, 0), а правим верхнім - у точці (W, H) та відрізки, паралельні осям координат. Відрізки задаються координатами своїх кінців. Ці відрізки розрізають прямокутник на кілька частин (можливо одну). Потрібно визначити їх площу. Відрізки можуть перетинатися, накладатися та вироджуватись у крапку. Усі координати - цілі числа за модулем не перевищують 10000.

# Вхідні дані

У першому рядку вхідного файлу INPUT.TXT вказуються числа W та H (1 ≤ W, H ≤ 10000). У другому рядку N (0 ≤ N ≤ 50) – кількість відрізків. Далі в N рядках через пропуск вказуються числа Ai, Bi, Ci, Di - координати кінців i-го відрізка: (Ai, Bi) та (Ci, Di).

# Вихідні дані

Вихідний файл OUTPUT.TXT повинен містити послідовність позитивних чисел

- Площі областей, записані в порядку не зростання.

# приклад

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 33 31311 1242 2 0 2 8 | 5211 |

## Завдання 8

В одній із найбільших IT-компаній вже багато років діє локальна мережа, до якої входять N комп'ютерів, послідовно пронумерованих цілими числами від 1 до N. Для з'єднання комп'ютерів використовується провід спеціального типу. За допомогою одного дроту можна безпосередньо з'єднати рівно два комп'ютери, причому з'єднання дозволяє передавати дані у двох напрямках. Кожен комп'ютер має два порти, тому може бути з'єднаний безпосередньо лише з двома іншими комп'ютерами. Мережа організована таким чином, що якщо два комп'ютери не з'єднані безпосередньо дротом, передача даних може здійснюватися через проміжні комп'ютери. Шляхом передачі

*даних*між комп'ютерами з номерами V1 та VK називається така послідовність комп'ютерів V1, V2, … , VK-1, VK, що для будь-якого 2 ≤ i ≤ K існує пряме з'єднання між комп'ютерами Vi-1 та Vi. Довжиною шляху передачі вважається кількість використовуваних проміжних комп'ютерів. Мінімальним шляхом передачі між комп'ютерами Vi і Vj є такий шлях передачі, довжина якого мінімальна з можливих.

Для наведеного нижче прикладу довжина мінімального шляху передачі даних між комп'ютерами 1 і 8 дорівнює 2, а між комп'ютерами 2 і 7 дорівнює 1 між комп'ютерами 2 і

4 дорівнює 0.

Якщо немає жодного шляху передачі між будь-якими двома комп'ютерами, то цьому випадку довжина мінімального шляху передачі між цими комп'ютерами вважається рівної нулю. Для наведеного вище прикладу між комп'ютерами 1 і 4 немає шляху передачі даних, тому довжина мінімального шляху передачі даних дорівнює 0.

Для з'ясування ефективності локальної мережі необхідно визначити коефіцієнт зв'язності, значення якого дорівнює сумі довжин мінімальних шляхів передачі між усіма парами комп'ютерів. Пари, що відрізняються порядком прямування елементів, вважаються різними, тобто пара (1, 2) і (2, 1) вважаються різними.

Вашим завданням є визначити значення коефіцієнта зв'язку для заданої локальної мережі.

# Вхідні дані

Перший рядок вхідного файлу INPUT.TXT містить два цілих числа N та M – число комп'ютерів та число прямих з'єднань відповідно (2 ≤ N ≤ 105, 1 ≤ M ≤ 106). Числа розділені поодиноким пропуском.

Кожен із наступних M рядків містить опис одного прямого з'єднання. Кожне пряме з'єднання описується за допомогою двох цілих чисел Xi та Yi (1 ≤ Xi, Yi ≤ N; Xi ≠ Yi), розділених пробілом – номери відповідних комп'ютерів, з'єднаних

безпосередньо. Відомо, що Xi ≠ Xj або Yi ≠ Yj, якщо i ≠ j. Рядки у файлі нумеруються послідовно, починаючи з одиниці.

# Вихідні дані

Єдиний рядок вихідного файлу OUTPUT.TXT повинен містити одне ціле число –

коефіцієнт зв'язку локальної мережі.

# Приклади

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 21 1 2 | 0 |
| 2 | 44 12 23 43 1 4 | 4 |
| 3 | 87 13 36 42 74 25 57 6 8 | 12 |

## Завдання9

Баштою називається вираз виду

...як

*a*23

*a*

*a*

*a*01

, де

*k*,Тобто послідовне

зведення у ступінь чисел a0, ..., ak. Зазначимо, що операція зведення у ступінь

виконуєся справа наліво, тобто. вираз abc обчислюється як abc.

Аня нещодавно вивчила алгоритми сортувань, і тепер вона вміє сортувати цілі числа, речові числа та навіть рядки. Дізнавшись про це, Андрій запропонував їй написати програму для сортування веж. Як це зробити?

Аня звикла вчитися на прикладах, і тому їй потрібна ваша програма, що працює. Напишіть програму, яка за заданими у вхідному файлі вежами обчислювала б порядок, у якому їх потрібно поставити, щоб вони були розташовані за зростанням.

# Вхідні дані

У першому рядку вхідного файлу INPUT.TXT задається число веж N (1 ≤ N ≤ 50000). Далі йдуть N рядків, кожен з яких містить одну вежу у форматі ki ai0 ai1 ... aiki, тут усі числа розділені пробілами. Кожне з aij - ціле число в межах від 1 до 99, крім

того, 1 ≤ ki ≤ 9.

Відомо, що серед веж у вхідному файлі немає рівних. Зауважте, що значення у веж

можуть бути дуже великі - наприклад, навіть число речовий тип.

222

22

65536не міститься в жодній

# Вихідні дані

У вихідний файл OUTPUT.TXT виведіть перестановку номерів веж b1, b2, ... , bN у такому

порядку, якщо взяти спочатку вежу з номером b1, потім із номером b2, тощо., вони виявляться розташованими порядку зростання.

# приклад

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 104 2 2 2 2 2122 132 123 3 2 2 2 22222 133 3 3 3 3 32433 2 2 3 4 | 2 4 3 6 7 5 9 10 1 8 |

## Завдання 10

У тривимірному просторі з прямокутною декартовою системою координат знаходяться n абсолютно пружних кульок. Для кожного з них відомі: маса mi, радіус ri, координати xi, yi, zi та вектор швидкості vi = (vix, viy, viz) у початковий момент часу. Необхідно розрахувати їх координати і швидкості після T секунд від початкового моменту часу. Кулі взаємодіють лише при зіткненнях, інших взаємодій між ними немає.

Врахуйте, що кулі можуть стикатися, і їхнє зіткнення описується законами збереження

*iy*

енергії та імпульсу, тобто зберігається величина

*n*

2

*v*

* *v*



*ix*

*i*

2 2

, а також вектор

*iz*

.

Зауважимо, що при зіткненні куль їх швидкості можуть змінитися тільки на вектор, паралельний прямий центри, що з'єднує їх в момент зіткнення.

# Вхідні дані

Перший рядок вхідного файлу INPUT.TXT містить натуральне число n (1 ≤ n ≤ 50) - кількість куль. Наступні n рядків вхідного файлу містять опис початкового стану куль, (i+1)-ий рядок містить 8 розділених пробілами дійсних чисел, не більше ніж з трьома знаками після коми: mi, ri, xi, yi, zi, vix, viy, viz .

Останній рядок містить ціле число T (0 ≤ T ≤ 100) - час, стан системи після якого треба розрахувати. Всі проекції швидкостей задані в метрах на секунду, всі радіуси та координати центрів – у метрах, всі маси – у кілограмах, а час T задано у секундах. Гарантується, що вхідні дані такі, що у кожному зіткненні беруть участь рівно 2 кулі. Усі числа у вхідному файлі не перевищують 100 по абсолютній величині. Початкове положення куль таке, що вони не торкаються один одного і не перетинаються. Маси та радіуси всіх куль суворо позитивні.

# Вихідні дані

У вихідний файл OUTPUT.TXT виведіть n рядків. На i-му рядку виведіть 6 дійсних чисел з точністю не менше ніж 3 знаки після десяткової точки: x-координату центру i-ої кулі, y-координату центру i-ої кулі, z-координату центру i-ої кулі, проекцію її швидкості на вісь Ox, проекцію його швидкості на вісь Oy, проекцію його швидкості на вісь Oz після T секунд. Координати виводьте у метрах, проекції швидкостей – у метрах за секунду.

# Приклади

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 31.0 1.0 0.0 0.0 0.0 1.00 1.0 1.02.0 2.0 10.0 10.0 10.0 -1.0 -1.0 -1.03.0 3.0 20.0 20.0 20.0 -0.5 -0.5 -0.5100 | -155.643 -155.643 -155.643 -1.667 -1.667 -1.667-47.028 -47.028 -47.028 -0.667 -0.667 -0.66726.566 26.566 26.566 0.167 0.167 0.167 |
| 2 | 21.01.0000000 1.01.01000-100 9 | -1.0000.0000.000-1.0000.0000.000 2.000 0.000 0.000 0.000 0.000 0.000 |

## Завдання 11

Рятувальники розробляють нову систему дій у разі виникнення аварійних ситуацій на хмарочосах. У цих випадках важливо визначати, які частини будівлі стоять стійко, а які ні.

На даний момент використовується наступна двовимірна модель. Будова описується як прямокутник, що складається з однакових блоків квадратної форми. Передбачається, що при аварії деякі блоки повністю руйнуються, а решта залишаються непошкодженими. Будемо називати сегментом безліч блоків, таких що з будь-якого можна

дійти до будь-якого іншого, якщо дозволяється переходити з блоку до блоку, що має з ним спільну сторону.

Вважається, що сегмент стоїть стійко, якщо один із його блоків стикається з нижньою стороною прямокутника.

Сегменти з блоків, які стоять нестійко, провалюються вниз до тих пір, поки якийсь із блоків сегмента не стикається з нижньою стороною прямокутника або з блоком стійкого сегмента. Після цього сегмент так само вважаються стійким.

За даними, які блоки збереглися, потрібно визначити положення верхнього блоку в кожному вертикальному стовпці.


# Вхідні дані

Перший рядок вхідного файлу INPUT.TXT містить одне ціле число m (1 ≤ m ≤ 10 000). За нею слідують m рядків, що описують вертикальні стовпці. Опис проводиться в такому порядку: спочатку записано число ai фрагментів цього вертикального стовпця, що залишилися цілими, за яким слідують 2ai чисел l(1)i, r(1)i, l(2)i, r(2)i, …, l( ai)i, r(ai)i, що задають нижні та верхні межі уцілілих фрагментів. При цьому 1 ≤ l(1)i ≤ r(1)i , l(2)i ≤ r(2)i , . . . , l(ai)i ≤ r(ai)i

≤ 106, для всіх допустимих i та j виконано r(j) < l(j+1) - 1. Сума всіх a не перевищує 100 000.

ii i

# Вихідні дані

У вихідний файл OUTPUT.TXT виведіть m чисел - висоти, на яких знаходиться верхній блок у відповідному вертикальному стовпці.

# приклад

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 82 1 1 4 62 1 1 6 62 1 4 6 6 | 5 5 5 5 6 6 6 1 |

2 1 1 6 6

3 1 1 4 6 8 8

2 1 1 8 8

2 1 3 7 8

1 6 6

## Задача1 2

Не всі в Тентурі мають право носити малинові штани, і звичайно, не всі володіють пепелацем з гравіцапою, тому для більшості жителів проблема переміщення між планетами була нерозв'язною. Але з деяких пір один заповзятливий чатланін з планети Плюк вийшов на ринок пасажирських перевезень, і за трохи чатлів, готовий перевозити охочих із планети на планету. Рейс починається з планети Плюк, що включає кількох інших планет, і завершується там же, на планеті Плюк. Однак під час підготовки рейсу виникли несподівані проблеми. Наприклад, якщо чатланін з планети Плюк хоче потрапити на планету, яка є в передостанній рейсі – йому мимоволі доведеться відвідати всі планети, які знаходяться в рейсі між планетою Плюк і його точкою призначення. Очевидно, що частина планет у цьому списку можуть виявитися пацацькими. Але кожен чатланін зобов'язаний носити цак на планеті пацакської і, навпаки, кожен пацак повинен носити цак на планеті чатланської. (Цак - дзвіночок для носа, відзнака для відносно нижчої касти на цій планеті). А процедура носіння цака принизлива у всіх сенсах цього слова.

Оскільки дане бюро подорожей поки що не має представництв на інших планетах, перевезення здійснюється тільки з планети Плюк на якусь іншу або з іншої планети на Плюк. Завдання планування рейсу спрощується – можна відвідувати планети у довільному порядку (але не можна відвідувати ту саму планету двічі – у дорозі може закінчитися луц). Необхідно обчислити такий порядок відвідування планет, у якому надягати цак на проміжних планетах доведеться мінімальну кількість разів.

# Вхідні дані

Перший рядок вхідного файлу INPUT.TXT містить натуральне число N ≤ 22 – кількість планет, що обслуговуються цим рейсом (не рахуючи планети Плюк). N наступних рядків містять інформацію про планети, в наступному вигляді: тип планети (латинська заголовна літера С - чатланська, P - пацакська), кількість чатлан, що йдуть до цієї планети з Плюка, кількість пацаків, що йдуть до цієї планети з Плюка, кількість чатлан, з цієї планети на Плюк, кількість пацаків, з цієї планети на Плюк. Усього пасажирів

≤ 1000.

# Вихідні дані

У вихідний файл OUTPUT.TXT виводиться, скільки разів доведеться надягати цак за оптимального маршруту, потім порядок відвідування планет через прогалину. Планети, перелічені у вхідному файлі, нумеруються починаючи з одиниці, планета Плюк має номер нуль і завжди вказується у послідовності двічі – на початку та наприкінці послідовності. Якщо існують кілька оптимальних маршрутів – слід вибрати той, де планета з меншим номером відвідується раніше.

# Приклади

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 2C 1 4 5 2P 2 5 1 4 | 5 0 2 1 0 |
| 2 | 4З 3 0 0 0З 3 0 0 1З 3 0 0 1З 3 0 0 1 | 3 0 1 2 3 4 0 |

**Пояснення**

У першому тесті можливі два варіанти маршруту: 0 1 2 0, і 0 2 1 0. У першому варіанті при посадці на чатланській планеті 1 п'ятьом пацакам, які прямують транзитом до своєї планети 2, доведеться надіти цак, а при наступній посадці на пацацькій планеті 2, п'ятьом чатланінам, які сіли на планеті 1 і прямують до Плюка, також доведеться одягнути цак. Разом у першому варіанті маршруту цак одягається 10 разів. У варіанті 2 цак на проміжних посадках надягає 4 і 1 раз відповідно, всього 5 разів. Другий варіант кращий.

У другому тесті всі планети чатланські, тому одягати цак доведеться лише пацакам. З Плюка пацаки не вирушають, але прибувають із трьох різних планет по одному. Першою відвідується єдина планета, де пацак не заходить у пепелац, потім йдуть три планети з однаковим набором пасажирів - на другому кроці з трьох рівноцінних планет, що залишилися, вибирається планета номер 2 як має найменший номер, потім 3 і решта 4. Пасажир з останньої планети не має проміжних посадок, з передостанньою здійснює одну посадку і надягає цак 1 раз, і пасажир, що залишився, одягає цак на двох проміжних зупинках, разом цак одягається 3 рази.

## Завдання 13

Антон працює у міжгалактичній туристичній агенції. Досить часто йому доводиться прокладати шлях із однієї планети в іншу з використанням існуючих рейсів космічних кораблів. На жаль, кількість рейсів невелика, тому пасажирам часто доводиться пересідати на проміжних планетах.

Антон зауважив, що деякі планети використовуються як проміжні частіше, ніж інші. Він вирішив провести дослідження – для кожної планети A він хотів би дізнатися, скільки існує пар різних планет (B, C), таких, що будь-який шлях із планети B на планету C проходить через планету A.

Допоможіть Антону!

# Вхідні дані

Перший рядок вхідного файлу INPUT.TXT містить два цілих числа: N і M - кількість планет і кількість рейсів космічних кораблів, відповідно (2 <= N <= 20 000, 1 <= M

<= 200000). Наступні рядки M описують рейси космічних кораблів. Кожен рейс пов'язує дві планети, і ним можна скористатися у будь-якому з двох напрямків. З будь-якої планети можна дістатися будь-якої іншої.

# Вихідні дані

У вихідний файл OUTPUT.TXT виведіть N цілих чисел – для кожної планети A виведіть кількість пар різних планет, таких що будь-який шлях з однієї планети на іншу проходить через A.

# приклад

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 79 12 13 14 15 16 17 23 45 6 7 | 18666666 |

## Завдання 14

Вася та Петя грають у наступну гру. Вася кладе на стіл два ряди карток. Перший ряд складається з N карток, кожної з яких написано деяке число ai. Другий ряд складається з N карток, кожної з яких написано деяке число bi.

Петі потрібно переупорядкувати картки першого ряду так, щоб на столі вийшло два однакових ряди карток. За одну секунду Петя може поміняти місцями i-у та (i+1)-у (1

<= i <= N-1) картки першого ряду.

Допоможіть Петі переупорядкувати картки, витративши мінімальний час.

# Вхідні дані

Перший рядок вхідного файлу INPUT.TXT містить ціле число N (1 <= N <= 105). Другий рядок містить N цілих чисел ai. Третій рядок містить N цілих чисел bi. Усі числа по абсолютній величині не перевищують 106.

# Вихідні дані

У вихідний файл OUTPUT.TXT виведіть кількість секунд, за які Петя зможе переупорядкувати картки необхідним чином. Якщо переупорядкувати картки необхідним чином неможливо, виведіть однину -1.

# Приклади

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 532345 5 4 3 2 3 | 7 |
| 2 | 81 2 1 2 1 2 1 22 1 2 1 2 1 2 2 | -1 |

## Завдання 15

Дано N відрізків на числовій прямій та M точок на цій же прямій. Для кожної з цих точок визначте, скільки відрізків вона належить. Точка x вважається відрізком, що належить, з кінцями a і b, якщо виконується подвійна нерівність min(a, b) <= x <= max(a, b).

# Вхідні дані

Перший рядок вхідного файлу INPUT.TXT містить два цілих числа N – число відрізків та M

- Число точок (1 <= N, M <= 105). У наступних N рядках по два цілі числа ai і bi – координати кінців відповідного відрізка. В останньому рядку M цілих чисел – координати точок. Усі числа у вхідному файлі не перевищують за модулем 109.

# Вихідні дані

У вихідний файл OUTPUT.TXT виведіть M чисел – для кожної точки кількість відрізків, у яких міститься.

# Приклади

|  |  |  |
| --- | --- | --- |
| **№** | **INPUT.TXT** | **OUTPUT.TXT** |
| 1 | 32 05 -32 710 1 6 | 2 0 |
| 2 | 13 -1010 -100 100 0 | 0 0 1 |