

C++ [4]

A. Никогда не играйте с незнакомцами

2 секунды, 64 мегабайта

Фёдор очень любит гулять неподалёку от Патриарших прудов. Во время очередной прогулки к нему подошёл незнакомец и предложил сыграть в карты. Фёдор согласился, и незнакомец объяснил ему правила.

Игра происходит колодой из карт на каждой из которых написано некоторое число — сила карты, известно, что правильный набор карт включает в себя N карт с силами $A, A + 1, A + 2, \dots, A + N - 1$, где A выбирается произвольным образом. На каждом ходу атакующий игрок выбирает некоторый набор карт из своей руки и выкладывает их на стол, защищающийся игрок либо выкладывает некоторый набор карт с суммарной силой строго большей чем сила карт атакующего игрока и отбивает эту атаку, либо берёт атакующий набор карт себе. После этого игроки добирают из колоды случайные карты, таким образом чтобы в руке каждого стало не менее 10 карт. Если защищающийся отбил атаку, то игроки меняются ролями и начинается новый раунд, побеждает игрок у которого в руке не осталось карт.

Фёдор крайне азартен и, к сожалению, проиграл кучу денег. Но он не готов признавать свои ошибки, по какой-то странной причине его оппонент оставил игральную колоду Фёдору, и Фёдор решил проверить, а действительно ли колода является правильной колодой для этой игры или же незнакомец его обманул.

Входные данные

В первой строке вам дано число N $1 \leq N \leq 10^5$ размер колоды. В следующей строке даны N чисел силы карт в колоде c_i ($1 \leq c_i \leq 10^9$).

Выходные данные

Если колода является корректной колодой для игры выведите "Deck looks good" без кавычек, в противном случае выведите "Scammed" без кавычек.

входные данные
3 1 2 3
выходные данные
Deck looks good

входные данные
3 10 100 1000
выходные данные
Scammed

B. Готовимся к зачету

2 секунды, 64 мегабайта

Очень сложно сдавать зачеты по информатике, если не готовиться к ним, а также если преподаватель пытается завалить любого студента на ровном месте. Так вышло с одной из групп первокурсников одного из факультетов одного из известных институтов. Однако все студенты известны своей способностью находить выход из любой ситуации. Обсудив все билеты после нескольких зачетов, они пришли к выводу, что при подготовке к зачету их преподаватель начинает уставать к последним билетам, поэтому самыми простыми билетами из составленных считались последние 2. Студенты подсмотрели, что преподаватель нумерует билеты целыми положительными числами по возрастанию. Также было замечено, что если есть билет с некоторым номером n , то не обязательно будет существовать билет с номером m , который меньше, чем n .

Так как билетов слишком много студентам пришлось вывести формулу, которой подчиняются номера билетов. Оказалось, что если преподаватель составил n различных билетов, то можно найти такое число a_0 , что номера всех билетов будут представлять собой последовательность $a_i = a_0 \cdot 5^i \pmod{10^9 + 7}; i \in [1, n]$.

Студенты еще не до конца научились программировать на языках программирования высокого уровня, поэтому попросили вас составить программу, которая определит номера двух самых простых билетов на ближайшем зачете.

Входные данные

В первой строке вам дано два числа n и a_0 ($2 \leq n \leq 10^8, 1 \leq a_0 < 10^9 + 7$), количество придуманных преподавателем билетов и число из формулы выше.

Выходные данные

Выведите 2 числа, упорядоченные по возрастанию, — ответ к задаче.

входные данные
4 1
выходные данные
125 625

входные данные
100 1
выходные данные
953952869 961070463

C. Быстрая сортировка

2 секунды, 128 мегабайт

Вам задан массив целых чисел, отсортируйте его.

Так как массив который требуется отсортировать слишком большой, задаваться он будет следующим образом: вам будут даны числа n, a, m, k и mod , с их помощью массив получается при помощи следующего кода:

```
vector<int> v(n);
for(int i = 0; i < n; ++i){
    v[i] = a;
    a = (a * m + k) % mod;
}
```

Из-за величины массива вывести его полностью так же невозможно поэтому для проверки корректности вашей сортировки от вас требуется вывести следующую величину посчитанную после сортировки массива:

$$\sum_{i=0}^{n-1} (i + 1) \cdot v[i] \pmod{10^9 + 7}$$

Входные данные

В первой строке вам дано число N ($1 \leq N \leq 5 \cdot 10^7$) — длина массива. В следующей строке вам даны числа a, m, k и mod ($0 \leq a, m, k < mod \leq 100333$) — числа определяющие генерацию массива.

Выходные данные

Выведите единственное число — ответ на задачу.

входные данные
3 1 1 1 10
выходные данные
14

В приведённом примере генерируется массив $\{1, 2, 3\}$, который уже является отсортированным и функция от него равняется $1 \cdot 1 + 2 \cdot 2 + 3 \cdot 3 = 14$.

D. Анаграммы

1 секунда, 256 мегабайт

Два слова называются анаграммами, если одно можно получить из другого путём перестановки букв. Например, "кот — ток" и "равновесие — своєнравие" являются парами анаграмм, а "кот — кит" не являются анаграммами. У Пети есть большая коллекция различных слов, так как она очень разрослась и её уже тяжело просмотреть всю за один раз, он хочет её как-нибудь уменьшить. Он решил, что меньше всего его коллекция пострадает если он оставит такое подмножество слов, что среди них не будет анаграмм и при этом невозможно будет больше взять ни одного слова из изначальной коллекции так, чтобы предыдущее условие не нарушилось. Помогите ему определить новый размер коллекции, чтобы он мог решить стоит ли всё это затевать или от этого его коллекция уменьшится недостаточно.

Входные данные

В первой строке дано единственное число N ($1 \leq N \leq 2 \cdot 10^5$) — количество слов в коллекции. В следующих N строках даны слова s_i ($1 \leq |s_i| \leq 20$) состоящие из маленьких латинских букв — изначальная коллекция Пети.

Выходные данные

Выведите единственное число, размер новой коллекции, которая останется у Пети, если он будет следовать своему плану.

входные данные
5 cat saw was silent listen
выходные данные
3

E. Объединение

2 секунды, 64 мегабайта

Василий создал онлайн игру в которой игроки могут объединяться в кланы ограниченного размера. Его игра стала довольно популярной и когда кланы стали достигать максимального размера, игроки попросили Василия увеличить максимальный размер кланов. К сожалению Василий не очень хороший программист и код написал так, что теперь невозможно изменить максимальный размер клана. Вместо этого он решил позволить кланам объединяться в союзы, которые не будут иметь ограничения на размер.

После внедрения новой системы игроки начали объединяться в союзы, при чём Василий замети один любопытный момент: объединялись всегда два наименее многочисленных клана. Так как игроки договариваются довольно медленно, Василий решил попробовать предсказать в каком порядке кланы будут объединяться далее. Реализуйте программу, которая определит порядок объединения кланов, если они будут действовать как предполагает Василий.

Входные данные

В первой строке вам дано единственное число N ($1 \leq N \leq 10^5$) — количество кланов в игре. В следующей строке через пробел даны сами размеры кланов a_i ($1 \leq a_i \leq 10^5$).

Выходные данные

В $N - 1$ строке выведите размеры кланов которые будут объединяться, сначала выводите размер меньшего клана, затем размер большего клана.

входные данные
6 1 2 3 4 5 6
выходные данные
1 2 3 3 4 5 6 6 9 12

F. Медианы

1 секунда, 64 мегабайта

Медиана массива — значение, которое окажется на позиции $\lfloor \frac{n-1}{2} \rfloor$ (где n длина массива), если его отсортировать (массив индексируется с нуля).

Ваша задача найти медианы для всех префиксов заданного массива.

Входные данные

В первой строке дано одно число n ($1 \leq n \leq 2 \cdot 10^5$) — размер массива. В следующей даны n целых чисел, не превосходящих по модулю 10^9 .

Выходные данные

Выведите n чисел — медианы всех префиксов заданного массива.

входные данные
4 4 3 2 1
выходные данные
4 3 3 2

$[4] \rightarrow [4] \rightarrow 4$
 $[4, 3] \rightarrow [3, 4] \rightarrow 3$
 $[4, 3, 2] \rightarrow [2, 3, 4] \rightarrow 3$
 $[4, 3, 2, 1] \rightarrow [1, 2, 3, 4] \rightarrow 2$

G. Размен строк

2 секунды, 64 мегабайта

Все граждане Берляндии хранят свои сбережения в строках состоящих из маленьких латинских букв и Теодор не исключение.

Сегодня утром Теодор проснулся и понял что неплохо было бы обменять одну из своих сберегательных строк на другую.

Согласно правилам Федерального Резерва Строк допускается обмен только определённых пар строк, попытка обмена неразрешённых пар преследуется по закону, однако гарантируется что последовательными обменами можно обменять любую строку на любую другу, при этом, возможно, не единственным способом. Если вы хотите обменять строку $A = a_0a_1a_2...$ на строку $B = b_0b_1b_2...$ и это допускается Резервом, то вам требуется заплатить цену, вычисляемую следующим образом.

- Меньшая строка дополняется символами '_' с конца.
- Каждому символу английского алфавита сопоставляется цена $p(c)$ по следующему правилу: $p('a') = 1, p('b') = 2, p('c') = 3, ... p('z') = 26$, а также вводится дополнительный символ '_' с ценой 0.
- Вы платите Резерву $\sum(p(b_i) - p(a_i))$, если цена получается отрицательной то соответствующую сумму Резерв выплачивает вам.

Теодор уже решил какую строку и на какую он хочет обменять, помогите ему понять сколько ему придётся минимально заплатить за последовательность обменов.

Входные данные

В первой строке вам дано количество пар которые позволено обменивать $N (1 \leq N \leq 10^4)$. В следующих N строках даны сами дозволенные пары. В последней строке даны строки которые Теодору необходимо обменять. Все строки не превышают по длине 100 символов.

Выходные данные

Выведите единственное число, минимальную цену за которую можно обменять первую строку на последнюю, возможно через некоторый набор промежуточных обменов. Если в итоге Резер останется должен Теодору то выведите соответствующее отрицательное число.

входные данные
1 timus codeforces codeforces timus
выходные данные
-11

входные данные
2 a b c b a c
выходные данные
2

Во втором примере Теодор сначала обменяет строку "a" на строку "b" заплатив 1, а затем обменяет строку b на строку c заплатив ещё 1. Иных вариантов обмена не существует, значит минимальная цена обмена строки "a" на строку "c" в заданных условиях 2.

Н. База данных

2 секунды, 64 мегабайта

Реализуйте базу данных, которая будет поддерживать следующий набор операций.

- "register <login> <password>" — регистрирует пользователя с логином "<login>" и паролем "<password>". Если желаемый логин уже занят, то выведите строку "login already in use", в противном случае создайте новый аккаунт и выведите строку "account created".
- "login <login> <password>" — даёт пользователю доступ к аккаунту "<login>" если указан верный пароль. Если требуемого пользователя не существует либо указанный пароль не верен выведите строку "wrong account info", если пользователь уже вошёл в систему с этим аккаунтом, то выведите строку "already logged in", в противном случае база должна запомнить что пользователь вошёл в систему и вывести строку "logged in".
- "logout <login>" — отбирает у пользователя доступ к аккаунту "<login>". Если требуемого пользователя не существует либо он не вошёл в систему выведите строку "incorrect operation", в противном случае база должна запомнить, что пользователь вышел из системы, и вывести строку "logged out".

Входные данные

Входный данные состоят из набора строк в указанным выше формате. Логины и пароли будут содержать только буквы латинского алфавита в верхнем и нижнем регистре, а так же цифры. Длина логина и паролей не будет превышать 20 символов. Общее количество запросов не будет превышать 10^5 .

Выходные данные

Для каждого запроса в отдельной строке выведите результат его выполнения.

входные данные
register vasya password login vasya passwodr login vasya password logout vasya register vasya qwerty register Vasya2017 qwerty logout vasya
выходные данные
account created wrong account info logged in logged out login already in use account created incorrect operation

11. Треугольники

2 секунды, 64 мегабайта

У Васи есть набор палочек различных длинн, помогите ему посчитать сколько различных невырожденных треугольников он сможет построить из этого набора. Треугольники считаются различными если в них используются различные по номеру палочки.

Входные данные

В первой строке вам дано число $N (1 \leq N \leq 700)$ — количество различных палочек у Васи. В следующей строке вам заданы числа $a_i (0 \leq a_i \leq 10^5)$ — длины палочек.

Выходные данные

Выведите количество различных треугольников которые Вася может составить из своего набора.

входные данные
5 1 2 3 4 5
выходные данные
3

входные данные
4 1 1 1 1
выходные данные
4

12. Быстрые треугольники

2 секунды, 64 мегабайта

У Васи есть набор палочек различных длинн, помогите ему посчитать сколько различных невырожденных треугольников он сможет построить из этого набора. Треугольники считаются различными если в них используются различные по номеру палочки.

Входные данные

В первой строке вам дано число $N (1 \leq N \leq 6000)$ — количество различных палочек у Васи. В следующей строке вам заданы числа $a_i (0 \leq a_i \leq 10^5)$ — длины палочек.

Выходные данные

Выведите количество различных треугольников которые Вася может составить из своего набора.

входные данные
5 1 2 3 4 5
выходные данные
3

входные данные
4 1 1 1 1
выходные данные
4

J. Внезапность

2 секунды, 64 мегабайта

Внезапно Вася обнаружил одну интересную задачу и решил дать её на тренировку, однако так как он проводит тренировки по всем правилам, ему нужно распечатать условие этой задачи для каждого участника тренировок. В его распоряжении имеются K ксероксов, которые он может использовать, чтобы копировать условия, каждый из которых копирует со своей скоростью. Определите как быстро Вася может распечатать необходимое количество условий. Копии можно делать как с оригинального условия так и с копий.

Входные данные

В единственной строке даны два числа N и K ($1 \leq N \leq 2 \cdot 10^{12}$, $1 \leq K \leq 10$) — необходимое количество копий и количество доступных ксероксов. В следующей строке через пробел даны K чисел t_i ($1 \leq t_i \leq 100$) — время создания копии на каждом из ксероксов.

Выходные данные

Выведите единственное число — минимальное время, необходимое для получения N копий условия задачи.

входные данные
1 2 2 3
выходные данные
2

входные данные
3 2 1 1
выходные данные
2

входные данные
5 2 1 2
выходные данные
4