

Задача А. ДНК роботов

Имя входного файла: `a.in`
Имя выходного файла: `a.out`
Ограничение по времени: 2 с
Ограничение по памяти: 256 мб

Последние достижения в технологии синтеза ДНК позволили провести эксперимент по созданию биороботов.

Для облегчения задачи создания ПО для управления роботами было принято решение, что их ДНК будет состоять из $M = 2^n$ символов для некоторого $n \geq 2$. Кроме этого, по техническим причинам это будет не обычная строка, а циклическая, то есть её можно начинать читать с любой позиции.

Одной из целей эксперимента является изучение мутаций биороботов. В результате продолжительных наблюдений было найдено много различных видов роботов. Для понимания процесса мутации учёным необходимо решить следующую задачу. Для ДНК двух роботов требуется определить коэффициент их похожести. Он вычисляется, как максимальное количество совпадающих символов при наилучшем совмещении этих ДНК. Чем больше символов совпадает, тем лучше совмещение.

Требуется написать программу, которая найдёт наилучшее совмещение двух ДНК.

Формат входного файла

В первой строке входного файла записано одно число M ($4 \leq M \leq 131072$). В следующих двух строках записаны ДНК двух роботов. Обе ДНК — строки, состоящие ровно из M символов из множества $\{‘A’, ‘C’, ‘G’, ‘T’\}$.

Формат выходного файла

В выходной файл выведите два числа — максимальное количество совпадающих символов и значение оптимального сдвига — неотрицательное количество символов второй ДНК, которые необходимо перенести из конца строки в её начало для достижения наилучшего совмещения.

Пример

<code>a.in</code>	<code>a.out</code>
16 ACGTACGTACGTACGT CGTACGTACGTACGTC	15 1

Задача В. Треугольник

Имя входного файла: `b.in`
 Имя выходного файла: `b.out`
 Ограничение по времени: 6 с
 Ограничение по памяти: 256 Мб

На плоскости расположено N ($3 \leq N$) точек. Из них случайным образом выбираются три точки, которые затем соединяются отрезками. Требуется определить математическое ожидание периметра получившегося треугольника, при условии, что каждое множество из трёх точек может быть выбрано с равной вероятностью, а получившийся треугольник может быть вырожденным.

Формат входного файла

В первой строке входного файла находятся два числа H и W ($1 \leq H, W \leq 700$). Далее следует H строк по W символов. j -й символ i -ой строки равен '1', если есть точка с координатами (i, j) , иначе в соответствующей позиции стоит символ '0'. Гарантируется, что во входных данных представлены как минимум три точки.

Формат выходного файла

В выходной файл выведите одно число — математическое ожидание периметра получившегося треугольника. Ответ должен отличаться от правильного не более, чем на 10^{-6} .

Примеры

<code>b.in</code>	<code>b.out</code>
<pre>11 20 10000000001000000000 00000000000000000000 00000000000000000000 00000000000000000000 00000000000000000000 00000000000000000000 00000000000000000000 00000000000000000000 00000000000000000000 00000000000000000000 00000000000000000000 00000000000000000000 00000000000000000000 10000000000000000000</pre>	34.142135624
<pre>3 3 101 010 101</pre>	5.794112550

Задача С. Монеты

Имя входного файла: `c.in`
Имя выходного файла: `c.out`
Ограничение по времени: 9 с
Ограничение по памяти: 256 Мб

Двое играют в следующую игру. На столе стоят две стопки монет. Игроки ходят по очереди. Во время хода игрок может взять любое ненулевое количество монет из любой стопки, либо взять некоторое одинаковое ненулевое количество монет из обеих стопок сразу. Проигрывает тот, кто не сможет сделать очередной ход.

Дано начальное количество монет в стопках. Требуется определить, кто выиграет при оптимальной игре.

Формат входного файла

В единственной строке входного файла записаны целые числа C_1 и C_2 ($1 \leq C_1 \leq 7^{7^7}$, $1 \leq C_2 \leq 7^{7^7}$) — количество монет в стопках.

Формат выходного файла

В выходной файл выведите `First`, если выигрывает первый игрок, или `Second`, если выигрывает второй.

Пример

<code>c.in</code>	<code>c.out</code>
1 1	First
1 2	Second
1 3	First

Задача D. Уравнение

Имя входного файла: `d.in`
Имя выходного файла: `d.out`
Ограничение по времени: 6 с
Ограничение по памяти: 256 Мб

Дано уравнение вида $X^N + Y^N \equiv Z^N \pmod{M}$.

Требуется для фиксированных N и M найти количество различных решений этого уравнения. Решением назовём такую тройку натуральных чисел (X, Y, Z) , что выполняется:

- $1 \leq X \leq Y < M$
- $1 \leq Z < M$
- $X^N + Y^N \equiv Z^N \pmod{M}$

Формат входного файла

В единственной строке входного файла записаны числа N и M ($1 \leq N \leq 7^7$, $1 \leq M \leq 7^7$).

Формат выходного файла

В выходной файл выведите одно число — ответ на задачу.

Примеры

<code>d.in</code>	<code>d.out</code>
1 3	2
2 4	5
3 5	8

Задача Е. Разворот битов

Имя входного файла: `e.in`
 Имя выходного файла: `e.out`
 Ограничение по времени: 2 с
 Ограничение по памяти: 256 Мб

Пусть $n = 2^s$, где $s \geq 0$ — целое число. Для каждого целого i от 0 до $n - 1$ определим значение a_i следующим образом. Пусть $b_{s-1} \dots b_1 b_0$ — запись числа i в двоичной системе счисления (при необходимости дополненная слева нулями до длины s). Переставим эти биты в обратном порядке ($b_0 b_1 \dots b_{s-1}$). Тогда значением a_i будет число, двоичной записью которого является $b_0 b_1 \dots b_{s-1}$.

Например, $s = 3, i = 3$, тогда двоичная запись будет выглядеть, как 011_2 , а запись в обратном порядке — 110_2 , следовательно, если $s = 3$, то $a_3 = 6$.

Требуется для заданного числа s быстро отвечать на запросы на подсчёт суммы $a_l + a_{l+1} + \dots + a_r$ по модулю 1000000007.

Формат входного файла

В первой строке входного файла записано целое число s ($0 \leq s \leq 31$). Во второй строке записано целое число k — количество запросов ($1 \leq k \leq 5 \cdot 10^5$). В следующих k строках содержатся сами запросы — пары целых чисел l, r ($0 \leq l \leq r < 2^s$).

Формат выходного файла

В выходной файл выведите k чисел — ответы на соответствующие запросы.

Примеры

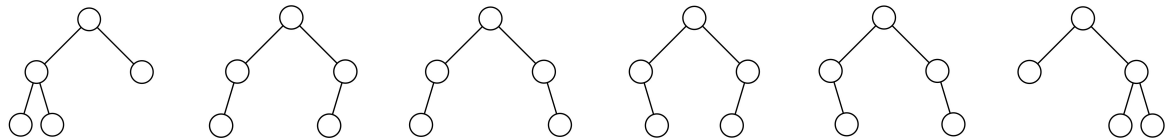
<code>e.in</code>	<code>e.out</code>
3 1 3 3	6
2 2 0 3 0 1	6 2
25 1 17 239	742367723

Задача F. AVL-деревья

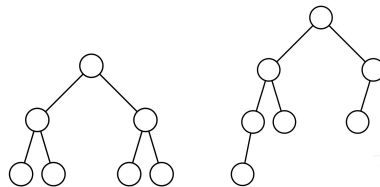
Имя входного файла: `f.in`
 Имя выходного файла: `f.out`
 Ограничение по времени: 5 с
 Ограничение по памяти: 256 Мб

AVL-дерево — сбалансированное по высоте двоичное дерево поиска: для каждой его вершины высота её двух поддеревьев различается не более чем на 1. AVL-деревья названы по первым буквам фамилий их изобретателей, Г. М. Адельсона-Вельского и Е. М. Ландиса.

Для фиксированного количества вершин может существовать несколько AVL-деревьев. Например, существует шесть AVL-деревьев, состоящих из пяти вершин.



Также деревья с одинаковым количеством вершин могут иметь различную высоту. Например, существуют деревья из семи вершин с высотами 2 и 3 соответственно.



Требуется по заданным n и h найти количество AVL-деревьев, состоящих из n вершин и имеющих высоту h . Так как ответ может быть очень большим, требуется найти остаток от деления искомого количества на 786433.

Формат входного файла

Во входном файле даны числа n и h ($1 \leq n \leq 65535$, $0 \leq h \leq 15$).

Формат выходного файла

Выведите одно число — остаток от деления количества AVL-деревьев, состоящих из n вершин и имеющих высоту h , на 786433.

Пример

<code>f.in</code>	<code>f.out</code>
7 3	16

786433 — простое число, $786433 = 3 \cdot 2^{18} + 1$.

Задача G. Многочлен

Имя входного файла: `g.in`
Имя выходного файла: `g.out`
Ограничение по времени: 7 с
Ограничение по памяти: 256 Мб

Многочлен $P(x) = a_0 + a_1x + a_2x^2 + \dots + a_{n-1}x^{n-1}$ задан набором коэффициентов a_0, a_1, \dots, a_{n-1} .

Требуется вычислить значения этого многочлена по модулю m для всех целых x от 0 до заданного числа k .

Формат входного файла

В первой строке входного файла записаны числа n , k и m ($1 \leq n \leq 2000$, $1 \leq k \leq 200000$, $1 \leq m \leq 10^9$).
Во второй строке записаны коэффициенты многочлена a_0, a_1, \dots, a_{n-1} — целые неотрицательные числа, не превосходящие 10^9 .

Формат выходного файла

В выходной файл выведите $k + 1$ число — остатки от деления значений $P(0), P(1), \dots, P(k)$ на m .

Примеры

<code>g.in</code>	<code>g.out</code>
2 4 239 17 3	17 20 23 26 29
3 5 11 5 8 1	5 3 3 5 9 4

Задача Н. Раздвоение

Имя входного файла: `h.in`
 Имя выходного файла: `h.out`
 Ограничение по времени: 2 с
 Ограничение по памяти: 256 Мб

Обозначим две последовательности действительных чисел $x(k)$ и $y(k)$. Определим последовательность комплексных чисел $z(k)$: $z(k) = x(k) + iy(k)$.

Пусть $FFT_N(k, z) = \sum_{n=0}^{N-1} z_n e^{2\pi i k n / N}$. Аналогичным образом определяются $FFT_N(k, x)$ и $FFT_N(k, y)$.

Требуется по вычисленным значениям $FFT_N(k, z)$ восстановить значения $FFT_N(k, x)$ и $FFT_N(k, y)$.

Формат входного файла

В первой строке входного файла записано целое число N ($1 \leq N \leq 2^{30}$, N является степенью двойки). Далее следуют целые неотрицательные числа A, B, C, D, E, F , не превосходящие 1000. Для экономии времени ввода значения $FFT_N(k, z)$ нужно будет вычислять по следующим формулам:

$$FFT_N(k, z).real = ((A + B \cdot k) \text{ xor } (C \cdot k)) \cdot 10^{-3},$$

$$FFT_N(k, z).imag = ((D + E \cdot k) \text{ xor } (F \cdot k)) \cdot 10^{-3},$$

где $FFT_N(k, z).real$ и $FFT_N(k, z).imag$ — действительная и мнимая части соответственно.

Затем дано число M — количество запросов ($1 \leq M \leq 10^5$). Далее следуют M целых чисел q_j ($0 \leq q_j < N$).

Формат выходного файла

В выходной файл выведите M строк. В j -ой строке — значения $FFT_N(q_j, x)$ и $FFT_N(q_j, y)$. Значения должны отличаться от правильных не более, чем на 10^{-4} .

Примеры

h.in	h.out
2 1000 0 0 0 0 0 2 0 1	1.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0
4 0 100 300 500 100 200 4 0 1 2 3	0.000 0.000 0.500 0.000 0.504 0.140 0.516 0.176 0.656 0.000 0.812 0.000 0.504 -0.140 0.516 -0.176
1048576 999 998 997 996 995 994 3 17 239239 2011	540.737 -1587.741 1589.778 539.689 2404.809 531.421 1359.578 1569.751 3678.277 -523.243 526.382 3664.887

Задача I. Уравнение 2

Имя входного файла: `i.in`
Имя выходного файла: `i.out`
Ограничение по времени: 4 с
Ограничение по памяти: 256 Мб

Дано уравнение вида $X^N + Y^N \equiv Z^N \pmod{M}$.

Требуется для фиксированных N и M найти количество различных решений этого уравнения. Решением назовём такую тройку натуральных чисел (X, Y, Z) , что выполняется:

- $1 \leq X \leq Y < M$
- $1 \leq Z < M$
- $X^N + Y^N \equiv Z^N \pmod{M}$

Формат входного файла

В единственной строке входного файла записаны числа N и M ($1 \leq N \leq 50000$, $1 \leq M \leq 50000$).

Формат выходного файла

В выходной файл выведите одно число — ответ на задачу.

Примеры

<code>i.in</code>	<code>i.out</code>
1 3	2
2 4	5
3 5	8

Задача J. Разворот битов 2

Имя входного файла: j.in
 Имя выходного файла: j.out
 Ограничение по времени: 2 с
 Ограничение по памяти: 256 Мб

Пусть $n = 2^s$, где $s \geq 0$ — целое число. Для каждого целого i от 0 до $n - 1$ определим значение a_i следующим образом. Пусть $b_{s-1} \dots b_1 b_0$ — запись числа i в двоичной системе счисления (при необходимости дополненная слева нулями до длины s). Переставим эти биты в обратном порядке ($b_0 b_1 \dots b_{s-1}$). Тогда значением a_i будет число, двоичной записью которого является $b_0 b_1 \dots b_{s-1}$.

Например, $s = 3, i = 3$, тогда двоичная запись будет выглядеть, как 011_2 , а запись в обратном порядке — 110_2 , следовательно, если $s = 3$, то $a_3 = 6$.

Требуется для заданного числа s быстро отвечать на запросы на подсчёт суммы $a_l + a_{l+1} + \dots + a_r$ по модулю 1000000007.

Формат входного файла

В первой строке входного файла записано целое число s ($0 \leq s \leq 25$). Во второй строке записано целое число k — количество запросов ($1 \leq k \leq 5 \cdot 10^5$). В следующих k строках содержатся сами запросы — пары целых чисел l, r ($0 \leq l \leq r < 2^s$).

Формат выходного файла

В выходной файл выведите k чисел — ответы на соответствующие запросы.

Примеры

j.in	j.out
3 1 3 3	6
2 2 0 3 0 1	6 2
25 1 17 239	742367723

Задача К. Дуэль

Имя входного файла: `k.in`
Имя выходного файла: `k.out`
Ограничение по времени: 3 с
Ограничение по памяти: 256 Мб

Двое дуэлянтов решили выбрать в качестве места проведения поединка тёмную аллею. Вдоль этой аллеи растёт n деревьев и кустов. Расстояние между соседними объектами равно одному метру. Дуэль решили проводить по следующим правилам. Некоторое дерево выбирается в качестве стартовой точки. Затем два дерева, находящихся на одинаковом расстоянии от исходного, отмечаются как места для стрельбы. Дуэлянты начинают движение от стартовой точки в противоположных направлениях. Когда соперники достигают отмеченных деревьев, они разворачиваются и начинают стрелять друг в друга.

Дана схема расположения деревьев вдоль аллеи. Требуется определить количество способов выбрать стартовую точку и места для стрельбы согласно правилам дуэли.

Формат входного файла

Во входном файле содержится одна строка, состоящая из символов '0' и '1' — схема аллеи. Деревья обозначаются символом '1', кусты — символом '0'. Длина строки не превосходит 100000 символов.

Формат выходного файла

Выведите количество способов выбрать стартовую точку и места для стрельбы согласно правилам дуэли.

Примеры

<code>k.in</code>	<code>k.out</code>
101010101	4
101001	0

В первом примере возможны следующие конфигурации дуэли (стартовое дерево и деревья для стрельбы выделены жирным шрифтом): **101010101**, **101010101**, **101010101** и **101010101**.

Задача L. Произведение

Имя входного файла: 1.in
Имя выходного файла: 1.out
Ограничение по времени: 2 с
Ограничение по памяти: 256 Мб

Требуется найти произведение двух целых чисел.

Формат входного файла

В каждой из двух строк входного файла записано целое число, состоящее не более чем из 239000 цифр.

Формат выходного файла

В выходной файл выведите одно целое число — произведение этих чисел.

Примеры

1.in	1.out
2 2	4
-1 1	-1

Задача М. Сфера

Имя входного файла: `m.in`
Имя выходного файла: `m.out`
Ограничение по времени: 3 с
Ограничение по памяти: 256 Мб

Требуется расположить на поверхности сферы N точек так, чтобы минимальное из попарных расстояний между этими точками было максимально возможным.

Формат входного файла

В единственной строке входного файла записаны целые числа R ($1 \leq R \leq 1000$) и N ($2 \leq N \leq 4$) — радиус сферы и количество точек соответственно.

Формат выходного файла

В выходной файл выведите одно действительное число — максимально возможное значение минимального из попарных расстояний между этими точками.

Выводить следует первые 10000 символов ответа (включая разделитель целой и дробной частей) без округления.

Примеры

<code>m.in</code>	<code>m.out</code>
1 2	2.00000...000
11 3	19.0525...344

Примечание

Для экономии бумаги 9990 символов ответа в примерах заменены на ‘...’.