

Содержание

7.Base [3/3]	3
Задача 7A. Очередь [0.1 sec, 256 mb]	3
Задача 7B. Постфиксная запись [0.1 sec, 256 mb]	4
Задача 7C. Уставший профессор [1 sec, 256 mb]	5
7.Advanced [3/5]	6
Задача 7A. Проверка ПСП [0.1 sec, 256 mb]	6
Задача 7B. Дети знакомятся [0.1 sec, 256 mb]	7
Задача 7C. Скобочная последовательность [0.1 sec, 256 mb]	8
Задача 7D. Максимальное число [0.1 sec, 256 mb]	9
Задача 7E. Удалите скобки [0.1 sec, 256 mb]	10
7.Hard [0/3]	11
Задача 7A. Хорошие дни [0.1 sec, 256 mb]	11
Задача 7B. Все минимумы [0.1 sec, 256 mb]	12
Задача 7C. Хороша ли подстрока? [0.1 sec, 256 mb]	13

Общая информация:

Вход в констест: <http://contest.yandex.ru/contest/3042/>

Дедлайн на задачи: 9 дней, до 2016-10-29 23:59.

К каждой главе есть более простые задачи (base), посложнее (advanced), и сложные (hard).

В скобках к каждой главе написано сколько любых задач из этой главы нужно сдать.

Сайт курса: <https://compscicenter.ru/courses/algorithms-1/2016-autumn/>

Семинары ведут Сергей Копелиович (burunduk30@gmail.com, vk.com/burunduk1) и Алексей Кладов (aleksey.kladov@gmail.com).

В каждом условии указан таймлимит для C/C++.

Таймлимит для Java примерно в 2-3 раза больше.

Таймлимит для Python примерно в 6 раз больше.

C++:

Быстрый ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/fread_write_export.cpp.html Более подробно про ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/cpp_common.html

В некоторых задачах нужен STL, который активно использует динамическую память (set-ы, map-ы) переопределение стандартного аллокатора ускорит вашу программу:

<http://acm.math.spbu.ru/~sk1/algo/memory.cpp.html>

Java:

Быстрый ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/java/java_common.html

7.Base [3/3]

Задача 7А. Очередь [0.1 sec, 256 mb]

В очереди в магазин стоят люди. Человек i хочет купить товар a_i . Изначально в магазине ничего нет. Происходят события следующих типов:

1. В момент времени T поступил один экземпляр товара A .
2. В момент времени T в конец очереди встал человек, который хочет купить товар A .

Нужно промоделировать процесс и для каждого человека определить, сколько он будет стоять в очереди. Замечание: как только первый в очереди может купить то, что хочет, он сразу мгновенно покупает и уходит.

Формат входных данных

Число событий N , $1 \leq N \leq 10\,000$. Далее события в порядке возрастания времени T . Каждое событие описывается так: $Type\ T\ A$, где $Type$ — тип события. $1 \leq T \leq 60\,000$, $1 \leq A \leq 10\,000$

Формат выходных данных

Для каждого человека (в том порядке, в котором люди вставали в очередь) выведите, сколько человек простоял в очереди. Если он так и остался стоять, выведите -1.

Примеры

queue.in	queue.out
9	20 40 20 0 -1
2 10 1	
2 20 1	
1 30 1	
2 40 2	
1 50 2	
1 60 1	
1 70 3	
2 80 3	
2 90 1	

Замечание

Просто очередь...

Задача 7В. Постфиксная запись [0.1 sec, 256 mb]

В постфиксной записи (или обратной польской записи) операция записывается после двух операндов. Например, сумма двух чисел A и B записывается как $A B +$. Запись $B C + D *$ обозначает привычное нам $(B + C) * D$, а запись $A B C + D * +$ означает $A + (B + C) * D$. Достоинство постфиксной записи в том, что она не требует скобок и дополнительных соглашений о приоритете операторов для своего чтения.

Дано выражение в обратной польской записи. Определите его значение.

Формат входных данных

В единственной строке записано выражение в постфиксной записи, содержащее однозначные числа и операции $+$, $-$, $*$. Строка содержит не более 100 чисел и операций.

Формат выходных данных

Необходимо вывести значение записанного выражения. Гарантируется, что результат выражения, а также результаты всех промежуточных вычислений по модулю меньше 2^{31} .

Пример

postfix.in	postfix.out
8 9 + 1 7 - *	-102

Замечание

Указание. Используйте стек.

Задача 7С. Уставший профессор [1 сек, 256 mb]

Уставший профессор вечером играет в увлекательную игру.

Изначально на доске слева направо записаны целые числа a_1, a_2, \dots, a_n . Пока не уснет, профессор каждую секунду смотрит на числа, видит, что самое левое равно x , а самое правое равно y . Если x меньше, то профессор радуется, стирает слева x , а справа дописывает $(x + y) \bmod 2^{30}$. Иначе профессор очень расстраивается, стирает y , а слева дописывает $(y - x) \bmod 2^{30}$. Студенты подсчитали, что перед сном профессор успел сделать ровно k операций. Что было написано на доске, когда он наконец заснул? Для простоты можно считать, что доска в обе стороны бесконечна.

Формат входных данных

На первой строке n ($1 \leq n \leq 30\,000$) и k ($1 \leq k \leq 10^8$). На второй строке числа a_1, a_2, \dots, a_n ($0 \leq a_i < 10^9$) в порядке слева направо.

Формат выходных данных

На первой строке выведите все числа на доске после k операций. Выводить числа нужно в порядке слева направо.

Примеры

sleepgame.in	sleepgame.out
4 1 1 2 3 4	2 3 4 5
4 1000 1 2 3 4	1062488873 1072033429 1060433235 57573251
4 1 4 3 2 1	1073741821 4 3 2
4 2 4 3 2 1	5 1073741821 4 3

Замечание

Видите изученную структуру данных?

Можете потренироваться и написать свой. Стандартный тоже зайдёт.

7. Advanced [3/5]

Задача 7А. Проверка ПСП [0.1 sec, 256 mb]

Дана строка, состоящая из круглых, квадратных и фигурных скобок. Нужно проверить, является ли она правильной скобочной последовательностью.

Формат входных данных

Во входном файле записана скобочная последовательность длиной не более 10 000 символов.

Формат выходных данных

Выведите YES, если скобочная последовательность является правильной, и NO в противном случае.

Примеры

check.in	check.out
([] ()	YES
([]	NO

Замечание

Разобрана на семинаре.

Задача 7В. Дети знакомятся [0.1 сек, 256 mb]

Мальчики и девочки со всего лагеря собрались на самой большой поляне, встали в круг, и решили познакомиться. У каждого ребенка есть имя. В определенном порядке каждый ребенок кричит «меня зовут X слева от меня стоит L , справа от меня стоит R », выходит из круга и убегает на полдник. Когда в круге остается 3 человека, они перестают кричать и спокойно расходятся. Известны имена детей и порядок, в котором они выходили из круга. Восстановите, что они кричали.

Формат входных данных

На первой строке количество детей n ($4 \leq n \leq 10^5$). На второй строке даны имена детей s_1, s_2, \dots, s_n . Дети даны в порядке против часовой стрелки и занумерованы целыми числами от 1 до n . Имена состоят из букв латинского алфавита. Длина имен не более 10. Имена могут совпадать. В третьей строке даны $n - 3$ числа — номера детей, которые выходили из круга.

Формат выходных данных

Для каждого вышедшего из круга ребенка выведите имена L и R , которые он прокричал.

Примеры

meeting.in	meeting.out
9	A C
A B C D E F G H Masha	C E
2 4 7 5 3 1	F H
	C F
	A F
	Masha F

Замечание

Видите двусвязный циклический список?

Чтобы он быстро работал, придётся писать свой.

Кстати, на C++ можно написать список, не используя, указатели.

Задача 7С. Скобочная последовательность [0.1 sec, 256 mb]

Дана скобочная последовательность, заданная символами «[», «]», «{», «}», «(», «)». Выведите самую длинную её подстроку, являющуюся правильной скобочной последовательностью.

Внимание. Решение должно работать за $O(n)$.

Определение. Подстрока строки $s_1s_2s_3\dots s_n$ – подотрезок $s_is_{i+1}\dots s_j$ для некоторых i, j .

Формат входных данных

Входные данные содержат скобочную последовательность длины от 1 до 100 000.

Формат выходных данных

Выведите требуемую подстроку, возможно пустую. Если максимальных по длине правильных подстрок несколько, то выведите любую.

Примеры

brackets.in	brackets.out
(()()())	(()()())
]))[(())]([[(())]
[([[([])])([[]]([[(())][[]]][]))	(())[[]]] []
])([] []	

Замечание

Разобрана на семинаре.

Задача про стек. Решение почти не отличается от “проверки на правильность”.

Задача 7D. Максимальное число [0.1 sec, 256 mb]

Дано длинное число из n цифр от 1 до 9. Необходимо оставить в нём k цифр, чтобы получившееся число было как можно больше (иначе говоря, необходимо вычеркнуть из него $n - k$ цифр; при этом оставшиеся цифры не меняют своего относительного порядка).

Внимание. Ваше решение должно работать за $O(n)$ и не должно асимптотически зависеть от основания системы счисления (в данном случае — десяти).

Формат входных данных

Первая строка содержит число k ($k > 1$) — количество цифр, которое нужно оставить в числе. Вторая строка содержит само число длины n (где $k \leq n \leq 200\,000$).

Формат выходных данных

Выведите число из k цифр, получившееся в итоге.

Примеры

maxnum.in	maxnum.out
1 57975	9
2 57975	97
3 198765	987
2 3131	33
4 12321	2321

Замечание

Есть решение со стеком.

Есть решение с очередью с максимумом.

Задача 7E. Удалите скобки [0.1 sec, 256 mb]

Дана строка, составленная из круглых скобок. Определите, какое наименьшее количество символов необходимо удалить из этой строки, чтобы оставшиеся символы образовывали правильную скобочную последовательность.

Формат входных данных

Во входном файле записана строка из круглых скобок. Длина строки не превосходит 100 000 символов.

Формат выходных данных

Выведите единственное целое число — ответ на поставленную задачу.

Примеры

erase.in	erase.out
()()	2
))((5
((()))	0

7.Hard [0/3]

Задача 7А. Хорошие дни [0.1 sec, 256 mb]

Билл разрабатывает новую математическую теорию, описывающую человеческие эмоции. Его последние исследования посвящены изучению того, насколько хорошие и плохие дни влияют на воспоминания людей о различных периодах жизни.

Недавно Билл придумал методику, которая описывает, насколько хорошим или плохим был день человеческой жизни с помощью сопоставления дню некоторого неотрицательного целого числа. Билл называет это число *эмоциональной значимостью* этого дня. Чем больше это число, тем лучше этот день. Билл полагает, что значимость некоторого периода человеческой жизни равна сумме эмоциональных значимостей каждого из дней периода, помноженной на минимум эмоциональных значимостей дней этого периода. Эта методика отражает то, что период, который в среднем может быть весьма неплох, бывает испорчен одним плохим днем.

Теперь Билл хочет проанализировать свою собственную жизнь и найти в ней период максимальной значимости. Помогите ему это сделать.

Формат входных данных

Первая строка входного файла содержит число n — количество дней в жизни Билла, которые он хочет исследовать ($1 \leq n \leq 100\,000$). Оставшаяся часть файла содержит n целых чисел a_1, a_2, \dots, a_n , все в пределах от 0 до 10^6 — эмоциональные значимости дней. Числа во входном файле разделяются пробелами и переводами строки.

Формат выходных данных

В первой строке выходного файла выведите максимальную значимость периода жизни Билла. Во второй строке выведите два числа l и r , означающие, что значимость периода с l -го по r -й день (включительно) в жизни Билла была максимально возможной.

Примеры

feelgood.in	feelgood.out
6	60
3 1 6 4 5 2	3 5

Замечание

Разобрана на семинаре.

Задача 7В. Все минимумы [0.1 сек, 256 mb]

Дан массив целых чисел a_1, a_2, \dots, a_n .

Для каждого его подотрезка $[a_L..a_R]$ определим $F(L, R) := \min\{a_L, \dots, a_R\}$.

Найдите

$$\sum_{1 \leq L \leq R \leq n} F(L, R)$$

то есть сумму минимумов всех подотрезков.

Формат входных данных

Первая строка входных данных содержит натуральное число n ($1 \leq n \leq 100\,000$) — размер массива. Во второй строке через пробел заданы элементы массива, все числа целые от -10^6 до 10^6 .

Формат выходных данных

Выведите единственное число — сумму минимумов всех подотрезков массива a .

Примеры

minsum.in	minsum.out
1 5	5
2 -10 1	-19
4 1 2 3 4	20
5 -3 2 -4 1 -5	-52

Замечание

Разобрана на семинаре.

Задача 7С. Хороша ли подстрока? [0.1 sec, 256 mb]

Вам дана строка из круглых скобок. Нужно отвечать на запросы вида “есть ли у подстроки $[l..r]$ исходной строки циклический сдвиг, являющийся правильной скобочной последовательностью (ПСП)?”.

Напомним определение ПСП:

- Пустая строка – ПСП.
- Если s – ПСП, то $'(s)'$ – тоже.
- Если s_1, s_2 – ПСП, то $'s_1s_2'$ – тоже.

Формат входных данных

Строка s из круглых скобок длины от 1 до 10^6 .

На второй строке количество запросов q ($1 \leq q \leq 10^5$).

Следующие q строк содержат пары чисел l, r ($1 \leq l \leq r \leq |s|$).

Формат выходных данных

Для каждого запроса, если ответ да, выведите 1, иначе 0.

Примеры

isgood.in	isgood.out
()())(111100
6	
1 2	
1 4	
5 6	
3 6	
1 5	
2 5	