Задача A. Космоплитка

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В 3030-м году хомяки решили благоустроить свой космический корабль. А именно уложить плитку на пол.

Пол имеет квадратную форму со стороной a метров. На него будет укладываться плитка прямоугольной формы, причём:

- стороны плитки должны быть параллельны сторонам пола;
- все плитки должны быть ориентированы одинаково;
- плитки будут укладываться вплотную друг к другу;
- первая плитка будет уложена вплотную к «верхнему левому» углу пола.

Для лучшего понимания расположения плитки смотрите иллюстрацию к примеру.

В распоряжении хомяков есть:

- 3D-чертеж плитки прямоугольной формы с размерами a метров на b метров (толщина настолько мала, что ей можно пренебречь);
- 3D-принтер, который может напечатать любое количество плиток;
- ультратонкий и ультраточный лазер, которым можно резать плитку, не оставляя при этом никаких отходов;
- уменьшающий лазер, которым можно уменьшить любую плитку в любое целое число x раз, но значение x должно быть не менее 1 и не более k;
- безудержная энергия для укладки плитки описанным выше способом.

План у хомяков следующий:

- Выбрать число x;
- Напечатать минимальное необходимое количество плиток;
- Обрезать часть плиток так, чтобы с помощью получившихся кусков плитки (с учётом дальнейшего уменьшения) можно было покрыть весь пол. Каждую плитку можно резать не более одного раза, только по прямой линии, параллельной стороне длиной a, и если резать на расстоянии r метров от стороны длиной a, то получатся два куска плитки: с размерами a на r метров и a на $b - r$ метров. Полученные куски резать нельзя;
- Те куски плитки, которые в итоге будут уложены на пол, уменьшить в x раз (то есть и длина куска плитки уменьшится ровно в x раз, и ширина плитки уменьшится ровно в x раз);
- Уложить плитку описанным выше способом.

Хомяки экономные, но соблюдают симметрию рисунка. Поэтому если при разрезе плитки на два куска куски получились одинаковых размеров, то используются оба куска (для лучшего понимания смотрите иллюстрацию к примеру при $x = 3$).

Хомяки поручили вам выбрать x таким образом, чтобы суммарная площадь неиспользованных кусков плитки была как можно меньше. Какова же эта минимальная суммарная площадь?

Формат входных данных

В первой строке дано целое число a — длина стороны пола и первой стороны плитки ($1 \leqslant a \leqslant 2 \cdot 10^6$).

Во второй строке дано целое число b — длина второй стороны плитки ($1 \leqslant b \leqslant 2 \cdot 10^6$).

В третьей строке дано целое число k — максимальное значение параметра x для уменьшающего лазера ($1 \leqslant k \leqslant 2 \cdot 10^{18}$).

Формат выходных данных

Выведите единственное число — минимальную суммарную площадь неиспользованных кусков.

Система оценки

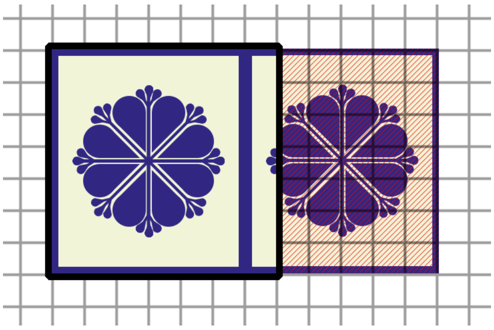
Подзадача	**Баллы**	**Доп. ограничения**	**Необх. Группы**	**Комментарий**
0	0	–	–	Тесты из условия.
1	20	$a, b, k \leqslant 200$; b — нечётное	–	–
2	15	$a, b, k \leqslant 200$	0–1	–
3	15	$a, b, k \leqslant 2\,000$	0–2	–
4	20	$k \leqslant 2 \cdot 10^6$	0–3	–
5	10	$k \leqslant 2 \cdot 10^{12}$	0–4	–
6	20	—	0–5	–

Примеры

стандартный ввод	стандартный вывод
7 6 4	21
11 8 4	0

Пояснение к примеру

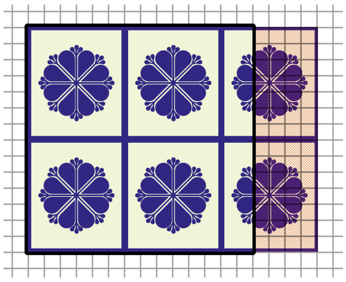
Иллюстрация к первому примеру для всех возможных x на следующей странице.



$x = 1$

Сторона маленького квадратика равна 1

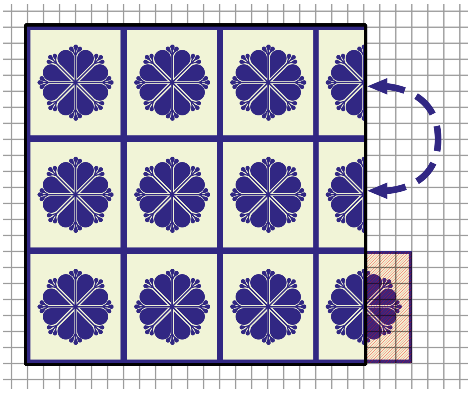
Искомая площадь равна $5 \cdot 7 = 35$



$x = 2$

Сторона маленького квадратика равна $\frac{1}{2}$

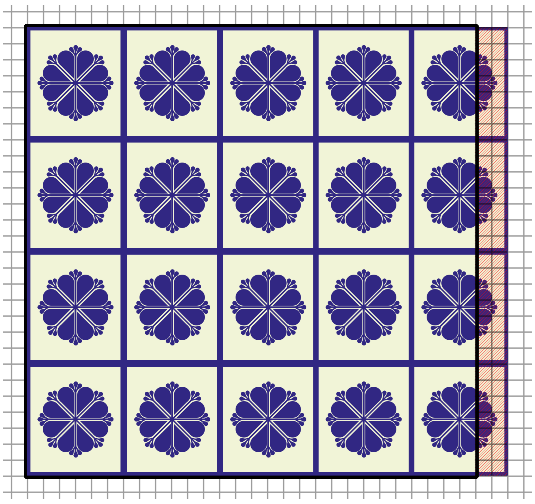
Искомая площадь равна $4 \cdot 14 = 56$



$x = 3$

Сторона маленького квадратика равна $\frac{1}{3}$

Искомая площадь равна $3 \cdot 7 = 21$



$x = 4$

Сторона маленького квадратика равна $\frac{1}{4}$

Искомая площадь равна $2 \cdot 28 = 56$

Заметьте, что величина маленького квадратика на рисунке не влияет на искомую суммарную площадь, так как к неиспользуемым кускам плитки не применяли уменьшающий лазер.

Задача B. Степенатор-2025

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Вам дан массив $a_1, a_2, \ldots, a_n$ из положительных целых чисел. У вас есть машина степенатор, которая может производить с массивом следующую операцию из двух частей:

1. Произвольным образом перемешать элементы в массиве $a_1, a_2, \ldots, a_n$
2. Заменить массив $a_1, a_2, \ldots, a_n$ на массив $a_1^{a_2}, a_2^{a_3}, \ldots, a_{n-1}^{a_n}$

Таким образом, после каждой операции длина массива уменьшается на один. Степенатор применит $n-1$ операцию, после чего в массиве a останется один элемент. Это число называется результатом работы степенатора.

Очевидно, результат работы степенатора не всегда определен однозначно. Чтобы изучить свойства степенатора, вам нужно узнать ответы на q вопросов вида: может ли результат работы степенатора нацело делиться на число x?

Формат входных данных

Первая строка входных данных содержит два целых числа n, q ($2 \leqslant n \leqslant 2 \cdot 10^5, 1 \leqslant q \leqslant 2 \cdot 10^5$) — длину массива a и количество запросов.

Вторая строка входных данных содержит n целых чисел $a_1, a_2, \ldots, a_n$ ($1 \leqslant a_i \leqslant 10^6$) — элементы массива a.

Следующие q строк содержат запросы по одному в строке, каждый запрос состоит из единственного числа x ($1 \leqslant x \leqslant 10^6$).

Формат выходных данных

Для каждого запроса, выведите одну строку: `Yes` или `No`. Если результат работы степенатора может делиться на x, выводите `Yes`, иначе `No`.

Система оценки

Тесты к этой задаче состоят из восьми групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп.

Группа	Баллы	Доп. ограничения		Необх. Группы	Комментарий
		n, q	x		
0	0	–	–	–	Тесты из условия.
1	10	$n = 2$	–	–	–
2	11	$n = 3$	–	–	–
3	12	$n = 4$	–	–	–
4	9	$n \cdot q = 10^6$	x свободно от квадратов	–	–
5	12	$n \cdot q = 10^6$	–	4	–
6	16	–	x свободно от квадратов	4	–
7	30	–	–	0-6	–

[1]Свободное от квадратов число — это число, которое не делится на квадрат любого простого числа. Например, 6 и 10 являются свободными от квадратов числами, а 4 (делится на 2^2) и 18 (делится на 3^2) — нет.

Примеры

стандартный ввод	стандартный вывод
3 2 2 2 1 16 5	Yes No
7 20 1 2 3 4 5 6 7 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20	Yes Yes Yes Yes Yes Yes Yes Yes Yes No No Yes No No No Yes No Yes No No

Задача C. Зайчик 3.1

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	0.75 секунд
Ограничение по памяти:	192 мегабайта

Зайчик стоит перед лестницей из n ступенек. Ступеньки пронумерованы от 1 до n и можно считать, что зайчик стоит на ступеньке номер 0. Для лучшего понимания смотрите иллюстрацию к примеру.

За раз зайчик может прыгнуть только через «красивое» количество ступенек. Зайчик считает красивым только то количество ступенек, у которого в записи в троичной системе счисления не встречается цифра 1. Прыгать назад нельзя.

Например, зайчик может прыгнуть через 0 ступенек и подняться тем самым на одну ступеньку вверх, так как $0_{10} = 0_3$ и в троичной системе счисления не использовалась цифра 1 для записи числа, или через 8 ступенек и подняться на девять ступенек вверх ($8_{10} = 22_3$), но не через 7 ступенек ($7_{10} = 21_3$).

Сколькими способами зайчик может подняться наверх (на n-ю ступеньку)?

Формат входных данных

В первой строке ввода содержится единственное число n ($1 \leqslant n \leqslant 2 \cdot 10^6$).

Формат выходных данных

Выведите единственное число — количество способов подняться наверх. Так как это число может быть очень большим, выведите его остаток при делении на 998 244 353.

Система оценки

Подзадача	Баллы	Доп. ограничения	Необх. Группы	Комментарий
0	0	–	–	Тесты из условия.
1	30	$n \leqslant 2000$	0	–
2	15	$n \leqslant 2 \cdot 10^4$	0–1	–
3	15	$n \leqslant 2 \cdot 10^5$	0–2	–
4	25	$n \leqslant 10^6$	0–3	–
5	15	—	0–4	–

Примеры

стандартный ввод	стандартный вывод
7	10
5	4
10	36

Пояснение к примеру

Иллюстрация к первому примеру на следующей странице. В этом примере можно было делать прыжки через $0_{10} = 0_3$, $2_{10} = 2_3$ и $6_{10} = 20_3$ ступенек.

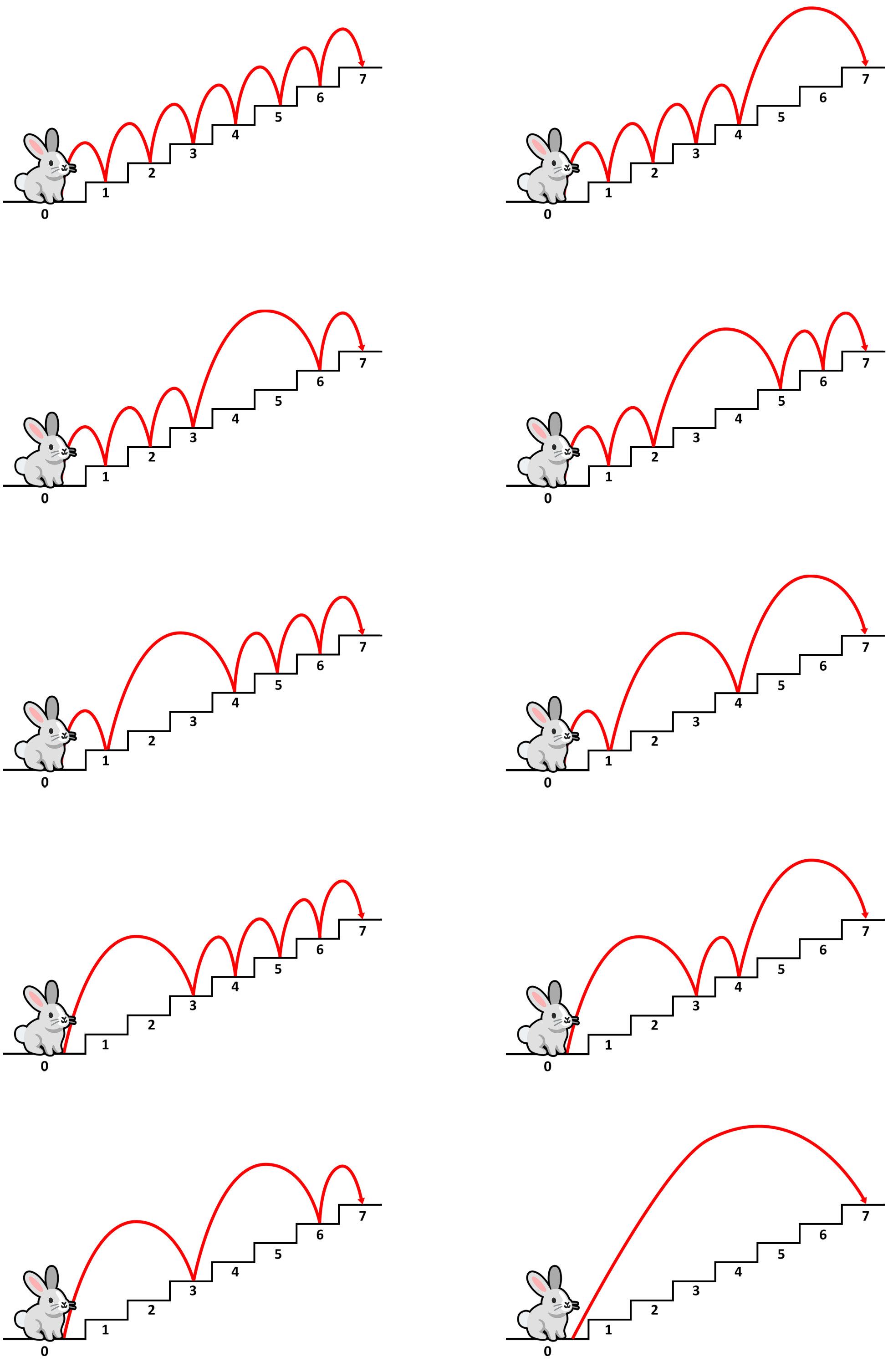


Иллюстрация к первому примеру

Задача D. Ремонт метро

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Станцию метро закрыли на m дней на ремонтные работы. Вы отвечаете за организацию работ на станции.

Всего есть n запланированных работ, которые можно выполнить за отведённое время. По оценкам рабочих, i-я работа может занять любое количество дней от l_i до r_i. Так как все работы технически сложные и оценить их время выполнения очень непросто, гарантируется, что выполнено $\frac{r_i}{l_i} \geqslant \alpha$ для всех i от 1 до n.

Ваша задача составить план ремонта: выбрать какое-то подмножество работ для выполнения. А чтобы вас не уволили за неэффективное распределение рабочих ресурсов, необходимо выполнить следующие условия:

- Вы должны гарантированно успеть завершить все работы за m дней. Другими словами, должно выполняться неравенство $r_{i_1} + r_{i_2} + \ldots + r_{i_k} \leqslant m$, в котором $i_1, i_2, \ldots, i_k$ — индексы работ, выбранных вами для выполнения.
- Пусть после того, как выполнение всех работ будет завершено, останется ещё d дней до открытия станции. Тогда не должно существовать такой невыполненной j-й работы, которую гарантированно можно успеть выполнить за d дней. Формально, для всех j должно выполняться неравенство $m - (l_{i_1} + l_{i_2} + \ldots + l_{i_k}) < r_j$, в котором $i_1, i_2, \ldots, i_k$ — индексы работ, выбранных вами для выполнения, а j — индекс **не** выбранной для выполнения работы.

Вы хотите узнать, сколько существует подмножеств работ, которые можно выбрать, чтобы выполнить оба этих условия. Так как ответ может быть достаточно большим, найдите его остаток при делении на $10^9 + 7$.

Формат входных данных

В первой строке дано два целых числа n и m — количество работ и общее время ремонтных работ ($1 \leqslant n \leqslant 5 \cdot 10^5$, $1 \leqslant m \leqslant 10^9$).

Во второе строке дано вещественное число α — параметр оценки работ ($1.5 \leqslant \alpha \leqslant 2$).

В каждой из следующих n строк задана пара чисел l_i, r_i ($1 \leqslant l_i \leqslant r_i \leqslant m$). Гарантируется, что $\frac{r_i}{l_i} \geqslant \alpha$.

Формат выходных данных

В единственной строке выведите количество возможных подмножеств работ для выполнения по модулю $10^9 + 7$.

Система оценки

Группа	Баллы	Доп. ограничения		Необх. Группы	Комментарий
		n	α		
0	0	–	–	–	Тесты из условия.
1	12	$n \leqslant 18$	$\alpha = 1.5$	0	–
2	13	$n \leqslant 25$	$\alpha = 1.5$	0–1	–
3	15	$n \leqslant 2000$	$\alpha = 2$	–	–
4	15	$n \leqslant 2000$	$\alpha = 1.5$	0–2	–
5	10	–	$\alpha = 2$	3	–
6	16	$n \leqslant 10^5$	$\alpha = 1.5$	0–2, 4	–
7	19	–	$\alpha = 1.5$	0–2, 4, 6	–

Примеры

стандартный ввод	стандартный вывод
4 10 1.5 2 3 1 2 2 5 3 7	1
7 20 2 1 3 2 4 6 12 9 20 5 12 5 13 6 13	2

Пояснение к примеру

В первом примере единственное подходящее подмножество состоит из первых трёх работ.

Задача E. MAX MEX MEX

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3.5 секунд
Ограничение по памяти:	256 мегабайт

Хомяк Ильнур подарил Алисе k корзинок с числами. На каждой корзинке написано число c_i.

Алиса может выполнить следующее действие несколько раз (возможно, ноль):

1. Выбрать i такое что $c_i > 0$.
2. Выбрать ровно одно число из i-й корзинки.
3. Выбранное число из i-й корзинки увеличить на 1, а число c_i уменьшить на 1.

То есть в i-й корзинке получится выполнить такое действие не более чем c_i раз.

Алисе нравится последовательность $0, 1, 2, \ldots$. Для каждой корзинки она умеет вычислять минимальный элемент, которого не хватает, чтобы продолжить такую последовательность.

Более конкретно — Алиса умеет находить $MEX^\dagger$ в каждой корзинке.

Алисе стало интересно — какое максимальное значение MEX всех MEX можно получить, выполнив несколько (возможно, ноль) указанных выше действий?

Обратите внимание на ограничение на k и $\sum n_i$ в последней подгруппе.

$^\dagger MEX$ (minimum excluded) массива — это наименьшее целое неотрицательное число, которое не принадлежит массиву.

Например:

MEX массива $[2, 2, 1]$ равен 0, так как 0 не принадлежит массиву.

MEX массива $[3, 1, 0, 1]$ равен 2, так как 0 и 1 принадлежат массиву, а 2 — нет.

MEX массива $[0, 3, 1, 2]$ равен 4, так как 0, 1, 2 и 3 принадлежат массиву, а 4 — нет.

Формат входных данных

В первой строке ввода содержится три числа — k ($1 \leqslant k \leqslant 10^5$), s ($1 \leqslant s \leqslant 10^6$), t ($1 \leqslant t \leqslant 2$).

В последующих $2 \cdot k$ строках содержится описание корзинок с числами.

Описание каждой корзинки зависит от значения t и выглядит следующим образом:

При $t = 1$:

Первая строка содержит два числа — n_i ($0 \leqslant n_i \leqslant s, \sum n_i = s$, т.е. сумма $n_i = s$), c_i ($0 \leqslant c_i \leqslant 10^{12}$).

Вторая строка содержит n_i чисел a_{i_j} ($-10^6 \leqslant a_{i_j} \leqslant 10^6$) — содержимое i-й корзинки.

При $t = 2$:

Первая строка содержит два числа — n_i ($0 \leqslant n_i \leqslant s, \sum n_i = s$), c_i ($0 \leqslant c_i \leqslant 10^{12}$).

Вторая строка содержит $2 \cdot n_i$ чисел — n_i пар чисел a_{i_j} ($-10^6 \leqslant a_{i_j} \leqslant 10^6$) b_{i_j} ($1 \leqslant b_{i_j} \leqslant 10^6$) — содержимое i-й корзинки.

Здесь b_{i_j} говорит о том, что число a_{i_j} повторяется b_{i_j} раз.

Обратите внимание, что если $n_i = 0$, то следующая строка будет пустой.

Формат выходных данных

Выведите единственное число — $MAX(MEX(MEX))$ всех корзинок с числами после выполнения необходимых действий.

Система оценки

Тесты к этой задаче состоят из девяти групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп.

Подзадача	Баллы	Доп. ограничения	Необх.Группы	Комментарий
0	0	–	–	Тесты из условия.
1	10	$t = 1, n_i \leqslant 15, k \leqslant 1000$	–	–
2	10	$t = 1, c_i = 0$	–	–
3	15	$t = 1, a_{i_j}$ различны в одной корзине	–	–
4	35	$t = 1$	1–3	–
5	30	$t = 2, k \leqslant 1000, s \leqslant 3 \cdot 10^4$	–	–

Примеры

стандартный ввод	стандартный вывод
4 5 2 1 5 -4 1 1 0 -1000000 1000000000 1 3 0 3 2 100 10 3 -4 2	4
4 8 1 1 5 -4 0 10 3 3 0 0 0 4 1000000000000 100 -1000000 -1 957	4

Замечание

В первом наборе исходных данных Алиса может применить такие действия:
Корзинка 1 $[-4] \Rightarrow [-3] \Rightarrow [-2] \Rightarrow [-1] \Rightarrow [0]$.
Корзинка 2 пустая [].
Корзинка 3 $[0, 0, 0] \Rightarrow [0, 0, 1] \Rightarrow [0, 1, 1] \Rightarrow [0, 1, 2]$
Корзинка 4 $[100, -1000000, -1, 957] \Rightarrow \ldots \Rightarrow [100, 0, 1, 957]$
Тогда $MEX_1 = 1, MEX_2 = 0, MEX_3 = 3, MEX_4 = 2$.
Следовательно $MEX([MEX_1, MEX_2, MEX_3, MEX_4]) = MEX([1, 0, 3, 2]) = 4$.
Во втором наборе исходных данных Алиса может применить такие действия:
Корзинка 1 $[-4] \Rightarrow [-3] \Rightarrow [-2] \Rightarrow [-1] \Rightarrow [0]$.
Корзинка 2 $[-1000000, -1000000, \ldots, -1000000]$ останется без изменений.
Корзинка 3 $[0, 0, 0] \Rightarrow [0, 0, 1] \Rightarrow [0, 1, 1] \Rightarrow [0, 1, 2]$
Корзинка 4 $[10, 10, 10, -4, -4] \Rightarrow \ldots \Rightarrow [10, 10, 10, 0, 1]$
Тогда $MEX_1 = 1, MEX_2 = 0, MEX_3 = 3, MEX_4 = 2$.
Следовательно $MEX([MEX_1, MEX_2, MEX_3, MEX_4]) = MEX([1, 0, 3, 2]) = 4$.