

1. Упорядочиванием массива A в рамках подмножества элементов $A[k[i]]$, где A - заданный массив, k - массив строго возрастающих индексов массива A , называется такая перестановка элементов массива A с индексами $k[i]$, что после перестановки элементы $A[k[i]]$ оказываются упорядоченными в требуемом смысле. Остальные элементы массива должны остаться на своих местах.

В файле `input.txt` записаны через пробел целые числа M , N и количество элементов массива A , далее сами элементы целочисленного массива A .

Упорядочить по возрастанию элементы целочисленного массива в рамках подмножества элементов, отличающихся от среднего арифметического всех элементов массива не более чем на M/N (под делением подразумевается обычное математическое деление).

Модифицированный массив должен быть выведен в файл `output.txt`: элементы массива должны быть разделены пробелом, количество элементов массива выводиться не должно.

Функция `main()` должна вернуть 0 в случае успешно решенной задачи и отрицательное число в случае невозможности решить задачу.

Считается, что решить задачу невозможно, если не удалось открыть файлы `input.txt` или `output.txt`; если файл `input.txt` содержит менее трех чисел; если массив в файле `input.txt` содержит меньше элементов, чем их заданное в файле количество; если $N=0$.

2. В файле `input.txt` записаны через пробел целые числа M , N и количество элементов массива A , далее сами элементы целочисленного массива A .

Каждый элемент в рамках подмножества целочисленных элементов, равных среднему арифметическому всех элементов массива, умноженному на M/N (имеется в виду обычное математическое деление), заменить на целую часть от полусуммы соседних элементов из всего массива. Элементы, не принадлежащие данному подмножеству, изменяться не должны.

Крайние элементы массива не менять.

Модифицированный массив должен быть выведен в файл `output.txt`: элементы массива должны быть разделены пробелом, количество элементов массива выводиться не должно.

Функция `main()` должна вернуть 0 в случае успешно решенной задачи и отрицательное число в случае невозможности решить задачу.

Считается, что решить задачу невозможно, если не удалось открыть файлы `input.txt` или `output.txt`; если файл `input.txt` содержит менее трех чисел; если массив в файле `input.txt` содержит меньше элементов, чем их заданное в файле количество; если $N=0$.

3. В файле `input.txt` записаны через пробел целые числа M , N и количество элементов массива A , далее сами элементы целочисленного массива A .

Каждый элемент в рамках подмножества целочисленных элементов, расположенных в диапазоне $[M, N]$, заменить на количество элементов из данного подмножества с большими индексами, которые меньше данного элемента, умноженного на (M/N) (имеется в виду обычное целочисленное деление из языка C в выражении (M/N)). Элементы, не принадлежащие данному подмножеству, изменяться не должны.

Модифицированный массив должен быть выведен в файл `output.txt`: элементы массива должны быть разделены пробелом, количество элементов массива выводиться не должно.

Функция `main()` должна вернуть 0 в случае успешно решенной задачи и отрицательное число в случае невозможности решить задачу.

Считается, что решить задачу невозможно, если не удалось открыть файлы `input.txt` или `output.txt`; если файл `input.txt` содержит менее трех чисел; если массив в файле `input.txt` содержит меньше элементов, чем их заданное в файле количество; если $N=0$.

4. В файле `input.txt` записаны через пробел целое число M , количество элементов массива A и далее сами элементы целочисленного массива A .

Каждый элемент в массиве A , для которого M -й бит (индексация с нуля) элемента совпадает с $M+1$ -м битом, заменить на количество единичных битов в представлении данного числа. Остальные элементы массива изменяться не должны.

Модифицированный массив должен быть выведен в файл `output.txt`: элементы массива должны быть разделены пробелом, количество элементов массива выводиться не должно.

Функция `main()` должна вернуть 0 в случае успешно решенной задачи и отрицательное число в случае невозможности решить задачу.

Считается, что решить задачу невозможно, если не удалось открыть файлы `input.txt` или `output.txt`; если файл `input.txt` содержит менее двух чисел; если массив в файле `input.txt` содержит меньше элементов, чем их заданное в файле количество; если $M < 0$ или $M > 30$.

5. В файле `input.txt` записаны через пробел целое число M , количество элементов массива A и далее сами элементы целочисленного массива A .

Рассмотрим подпоследовательности элементов исходного массива, определяемые следующими условиями: элементы подпоследовательности идут подряд в исходном массиве, четность количества единичных битов элементов совпадает с четностью числа M , для элементов подпоследовательности справедливо неравенство $A[i] > A[i+1]/2$ (имеется в виду обычное математическое деление), в каждой подпоследовательности содержится более одного элемента, каждую подпоследовательность нельзя расширить следующим или предыдущим элементом массива.

В каждой такой подпоследовательности заменить каждый элемент на количество элементов всего данного массива, отличающихся от среднего арифметического всех элементов массива не более чем на M . Элементы, не принадлежащие данным подпоследовательностям, изменяться не должны.

Модифицированный массив должен быть выведен в файл `output.txt`: элементы массива должны быть разделены пробелом, количество элементов массива выводиться не должно.

Функция `main()` должна вернуть 0 в случае успешно решенной задачи и отрицательное число в случае невозможности решить задачу.

Считается, что решить задачу невозможно, если не удалось открыть файлы `input.txt` или `output.txt`; если файл `input.txt` содержит менее двух чисел; если массив в файле `input.txt` содержит меньше элементов, чем их заданное в файле количество.

6. В файле `input.txt` записаны через пробел целое число M , количество элементов массива A и далее сами элементы целочисленного массива A , состоящего из неотрицательных элементов.

Рассмотрим подпоследовательности элементов исходного массива, определяемые следующими условиями: элементы подпоследовательности идут подряд в исходном массиве, M -й бит элемента (индексация с нуля) равен 0, для элементов подпоследовательности справедливо равенство $A[i] = A[i+1]/2$ (имеется в виду обычное математическое деление), в каждой подпоследовательности содержится более одного элемента, каждую подпоследовательность нельзя расширить следующим или предыдущим элементом массива.

В каждой такой подпоследовательности заменить каждый элемент на целую часть среднего арифметического элементов этой конкретной подпоследовательности. Элементы, не принадлежащие данным подпоследовательностям, изменяться не должны.

Модифицированный массив должен быть выведен в файл `output.txt`: элементы массива должны быть разделены пробелом, количество элементов массива выводиться не должно.

Функция `main()` должна вернуть 0 в случае успешно решенной задачи и отрицательное число в случае невозможности решить задачу.

Считается, что решить задачу невозможно, если не удалось открыть файлы `input.txt` или `output.txt`; если файл `input.txt` содержит менее двух чисел; если массив в файле `input.txt` содержит меньше элементов, чем их заданное в файле количество; если $M \leq 0$ или $M > 31$.

Функция `main()` должна вернуть 0 в случае успешно решенной задачи и отрицательное число в случае невозможности решить задачу.

Считается, что решить задачу невозможно, если не удалось открыть файлы `input.txt` или `output.txt`; если файл `input.txt` содержит менее двух чисел; если массив в файле `input.txt` содержит меньше элементов, чем их заданное в файле количество; если $M \leq 0$ или $M > 31$.