

Задача 5. «Парк аттракционов»

Имя входного файла:	attract.in
Имя выходного файла:	attract.out
Максимальное время работы на одном тесте:	1 секунда
Максимальный объем используемой памяти:	256 мегабайт
Максимальная оценка	100 баллов

В городе \square недавно построили парк аттракционов, в котором есть павильон игровых автоматов. Каждый из автоматов рассчитан на одного человека. В программе Всероссийской олимпиады планируется посещение этого павильона.

Перед организаторами встала сложная задача — составить расписание игры участников олимпиады на автоматах таким образом, чтобы каждый из N участников олимпиады смог поиграть на каждом из автоматов, и при этом автобус, увозящий участников из парка олимпиады, смог бы отправиться к месту проживания как можно раньше.

Время перемещения участников между автоматами, а также между автобусом и павильоном считается равным нулю. Каждый из участников в любой момент времени может как играть на автомате, так и ждать своей очереди, например, гуляя по парку. Для каждого из M ($M \leq N$) автоматов известно время игры на нём t_i ($1 \leq i \leq M$). Прервать начатую игру на автомате невозможно. Автобус привозит всех участников олимпиады в парк одновременно в нулевой момент времени.

Требуется написать программу, которая по заданным числам N , M и t_i определяет оптимальное расписание игры на автоматах для каждого из участников.

Формат входных данных

В первой строке входного файла содержатся два числа: N и M ($1 \leq M \leq N \leq 100$). Во второй строке заданы M целых чисел t_i ($1 \leq t_i \leq 100$), каждое из которых задаёт время игры на i -м автомате ($1 \leq i \leq M$). Числа в строке разделяются одиночными пробелами.

Формат выходных данных

В первой строке необходимо вывести одно число — минимально возможное время отправления автобуса из парка аттракционов. Далее необходимо вывести N расписаний игр на автоматах, по одному для каждого из участников. Каждое расписание описывается в $(M + 1)$ строках, первая из которых — пустая, а далее следуют M строк, описывающих автоматы в порядке их посещения этим участником. Посещение автомата описывается двумя целыми числами: номером автомата j ($1 \leq j \leq M$) и временем начала игры участника на этом автомате.

Примеры

attract.in	attract.out
2 1 2	4 1 0 1 2
3 2 2 1	6 1 0 2 2 1 2 2 4 2 0 1 4

Подзадачи и система оценки

Данная задача содержит пять подзадач. Для оценки каждой подзадачи используется своя группа тестов. Баллы за подзадачу начисляются только в том случае, если все тесты из этой группы пройдены.

Подзадача 1 (20 баллов)

$M = 1$, $1 \leq N \leq 100$, t_i лежит в пределах от 1 до 100.

Подзадача 2 (20 баллов)

Все t_i равны 1, $N = M$.

Подзадача 3 (20 баллов)

Все t_i равны 1, $N > M$.

Подзадача 4 (20 баллов)

Числа t_i лежат в пределах от 1 до 100, $N = M$.

Подзадача 5 (20 баллов)

Числа t_i лежат в пределах от 1 до 100, $N > M$.

Обратная связь

В течение тура можно не более 10 раз запросить результаты работы своей программы на тестах жюри. Запрос можно делать не чаще одного раза в 5 минут. Для каждого теста сообщается результат запуска программы на этом тесте.

В этой задаче можно выбрать, какое решение будет оцениваться. В этом случае баллы начисляются за лучшее решение из следующих:

- выбранного явно;
- последнего принятого на проверку решения.

Если выбор не сделан, то будет оцениваться лучшее решение из следующих:

- решений, по которым просмотрены баллы;
- последнего принятого на проверку решения.