

ЗАМЕЩЕНИЕ СТРАНИЦ

Description

Вам нужно написать программу, моделирующую работу страничной системы с алгоритмом замещения страниц LRU (Least Recently Used). Для реализации алгоритма LRU нужно хранить связный список страниц.

Суть алгоритма заключается в том, что последняя использовавшаяся страница находится в начале списка страниц, а страница, к которой дольше всего не обращались, находится в конце списка.

Вот пример работы страничной системы с алгоритмом LRU

В физической памяти есть 4 страничных блока.

Пусть есть последовательность обращений к страницам:

0 2 1 3 5 4 6 3 7 4 7 3 3 5 5 3 1 1 1 7 1 3 4 1

Смоделировав страничную систему получим таблицу.

Последовательность обращений	0	2	1	3	5	4	6	3	7	4	7	3	3	5	5	3	1	1	1	7	1	3	4	1
	0	2	1	3	5	4	6	3	7	4	7	3	3	5	5	3	1	1	1	7	1	3	4	1
		0	2	1	3	5	4	6	3	3	4	4	7	7	7	5	5	5	3	3	7	1	3	
			0	2	1	3	5	4	6	6	6	6	4	4	4	7	7	7	5	5	5	7	7	
				0	2	1	1	5	5	5	5	5	6	6	6	4	4	4	4	4	4	4	5	5
					0	2	2	1	1	1	1	1	1	1	1	6	6	6	6	6	6	6	6	6
						0	0	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Жирный контур отделяет 4 верхних элемента списка, которые соответствуют страницам загруженным в память. Страницы ниже контура выгружены на диск.

Input

Первая строка входного файла содержит целые N и M ($0 < N, M < 100$), где N – количество обращений к страницам, M – количество страничных блоков в физической памяти. Затем следуют N строк, где каждая строка соответствует номеру страницы из последовательности обращений.

Output

Вывести список страниц, который будет сформирован после работы страничной системы. Каждая строка выходного файла содержит номер страницы из списка и через пробел количество загрузок в память этой страницы.

Sample Input

24 4
0
2
1
3
5
4
6
3
7
4
7
3
3
5
5
3
1
1
1
7
1
3
4
1

Output for the Sample Input

1 2
4 2
3 1
7 1
5 2
6 1
2 1
0 1