

Задача А. Достижимые вершины

Имя входного файла: a4.in
Имя выходного файла: a4.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Задан неориентированный граф, нужно определить, какие вершины достижимы из заданной вершины S (находятся с ней в одной компоненте связности).

Формат входного файла

В первой строке записаны три числа N, M, S разделённые пробелами. N - количество вершин ($1 \leq N \leq 255$). M - количество ребёр. ($1 \leq M \leq 255 * 255$)

Далее идёт M строк. Каждая строка задаёт одно ребро и содержит 2 числа F, T . Ребро соединяет вершины F и T .

Формат выходного файла

Выведите все достижимные из S вершины в порядке возрастания разделённые пробелами.

Примеры

a4.in	a4.out
4 2 2 1 2 3 4	1 2
4 2 3 1 4 2 2	3
5 4 3 3 4 1 1 5 2 1 4	1 3 4

Задача В. Стек

Имя входного файла: b4.in
Имя выходного файла: b4.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Вам нужно промоделировать работу стека. Во входном файле написаны команды, после выполнения каждой команды вам надо выводить состояние стека.

Формат входного файла

В первой строке одно число $1 \leq N \leq 1000$ - количество команд. В каждой из N последующих строк по одной из команд.

- PUSH число - положить на вершину стека целое число и вывести состояние стека.
- POP - взять с вершины стека число, вывести его и вывести состояние стека.

Формат выходного файла

В каждой строке выходного файла вывести состояние стека (для PUSH) и и полученный элемент и состояние стека (для POP) как показано в примерах.

Примеры

b4.in	b4.out
4 PUSH 2 POP PUSH 1 POP	[2] 2 [] [1] 1 []
5 PUSH 3 PUSH 4 POP PUSH 2 POP	[3] [3, 4] 4 [3] [3, 2] 2 [3]
7 PUSH 1 PUSH 7 PUSH 3 POP POP PUSH 2 PUSH 9	[1] [1, 7] [1, 7, 3] 3 [1, 7] 7 [1] [1, 2] [1, 2, 9]

Задача С. Очередь

Имя входного файла: c4.in
Имя выходного файла: c4.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Вам нужно промоделировать работу очереди. Во входном файле написаны команды, после выполнения каждой команды вам надо выводить состояние очереди.

Формат входного файла

В первой строке одно число $1 \leq N \leq 1000$ - количество команд. В каждой из N последующих строк по одной из команд.

- PUT X - положить в начало очереди целое число и вывести состояние очереди. $0 \leq X \leq 10^9$.
- GET - взять из конца очереди элемент, вывести его и вывести состояние очереди.

Формат выходного файла

В каждой строке выходного файла вывести состояние очереди (для PUT) и и полученный элемент и состояние стека (для GET) как показано в примерах.

Примеры

c4.in	c4.out
5 PUT 3 PUT 4 PUT 3 GET PUT 6	[3] [4, 3] [3, 4, 3] 3 [3, 4] [6, 3, 4]
6 PUT 7 PUT 9 PUT 7 PUT 9 GET GET	[7] [9, 7] [7, 9, 7] [9, 7, 9, 7] 7 [9, 7, 9] 9 [9, 7]