

Имате дадено двоично дърво (BST) за търсене с N елемента и цяло число K . Изтрийте K -тия най-голям елемент и изведете стойностите на дървото подредени по нива от дясно на ляво (level order). Ако изтрият елемент има два наследника, елементът заемащ мястото на изтрития е най-лявото листо в дясното му поддърво.

Иска се да изведете първо корена после върховете на ниво 1 от ляво на дясно и после върховете от ниво 2 от ляво на дясно и тн.

Input Format

На първия ред от стандартният вход се въвеждат N и K .

Следват N на брой цели числа a_0, \dots, a_{N-1} - елементите, които трябва да се добавят в дървото. Родителят на елемента a_i е $a_{(i-1)/2}$.

Върхът, който съдържа стойността на индекс i в масива е родител на върховете със стойности съответно на индекси $2 \times i + 1$ и $2 \times i + 2$

Constraints

$$0 \leq N \leq 10^5$$

$$0 \leq a_i \leq 10^5$$

Output Format

Изведете $N - 1$ цели числа - останалите елементи в дървото, след премахването на K -тия най-голям.

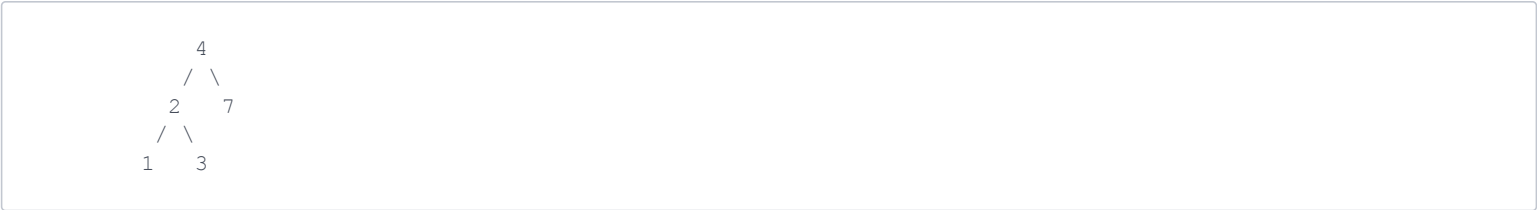
Sample Input 0

```
5 1
4 2 7 1 3
```

Sample Output 0

```
4 2 1 3
```

Explanation 0



1-вият най-голям елемент е 7 => след изтриването му дървото изглежда така:



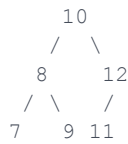
Sample Input 1

6 5
10 8 12 7 9 11

Sample Output 1

10 9 12 7 11

Explanation 1



5-тият най-голям елемент е 8 => след изтриването му дървото изглежда така:

