

Tree operations (part II)

<https://www.hackerrank.com/contests/sda-exam-27-01-19-/challenges/challenge-1796>

Използвайте вече написаните функции `add` и `print` и добавете следните нови функции:

`void remove(int X)` - ако в дървото имаме елемент `X` го изтрива (запазвайки структурата наредено двоично дърво. Упътване: ако възела няма наследник просто се премахва, ако има само един наследник то той отива на негово място, а ако има два наследника тогава се търси най-малкият по-голям от него). Не принтира нищо на стандартният изход.

`print_odd_layers()` - принтира елементите на дървото от нечетните му нива започвайки от корена и продължавайки с ниво 3, ниво 5 и т.н. (елементите от четните нива се прескачат).

Входен формат

`N` - брой на операциите. Следват имената на операциите като аргументите са разделени с интервал.

Ограничения

$1 < N < 200\,000$

Изходен формат

Изход спрямо изпълнените операции.

Примерен вход	Очакван изход
5 add 2 add 2 add 1 remove 2 print	1
8 add 1 add 2 add 4 add 6 add 7 add 5 print_odd_layers print	1 4 5 7 1 2 4 6 5 7

github.com/andy489