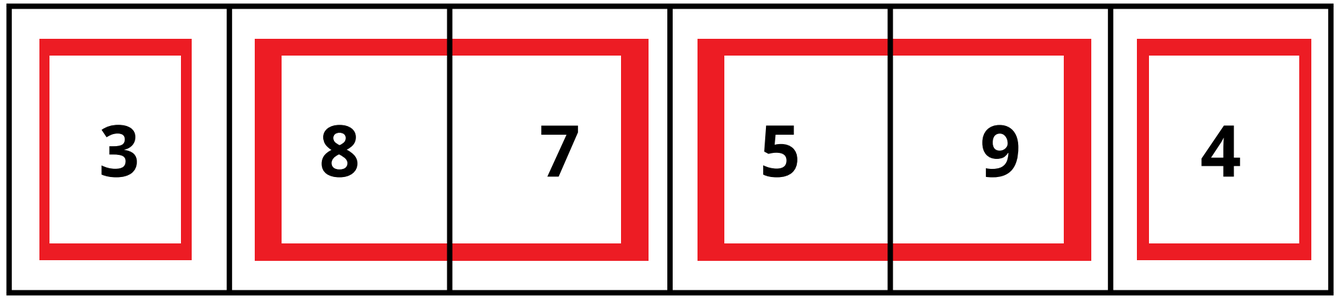
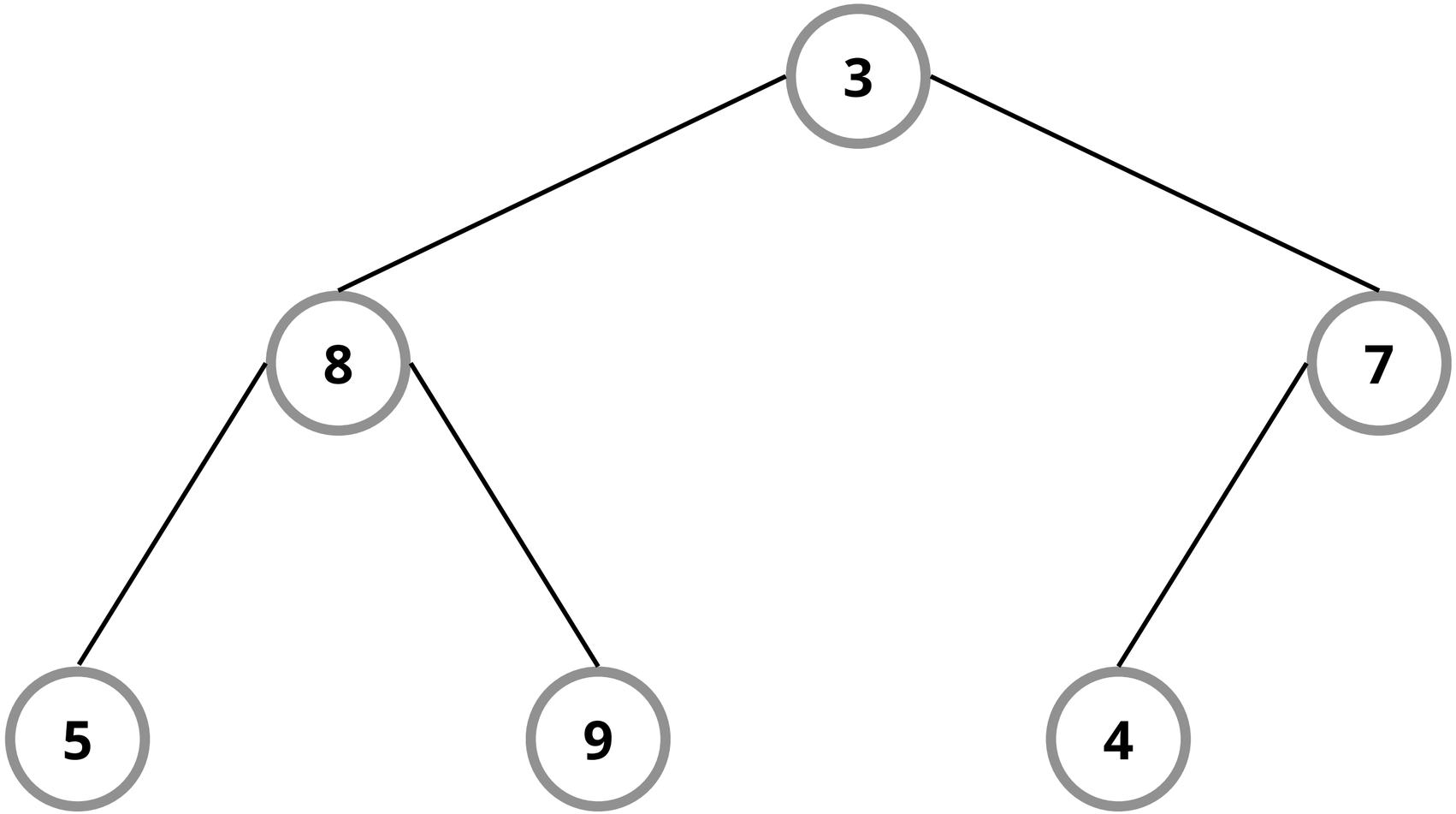
tentukan array yang akan disorting terlebih dahulu, array yang belum terurut untuk kasus ini



Array Input

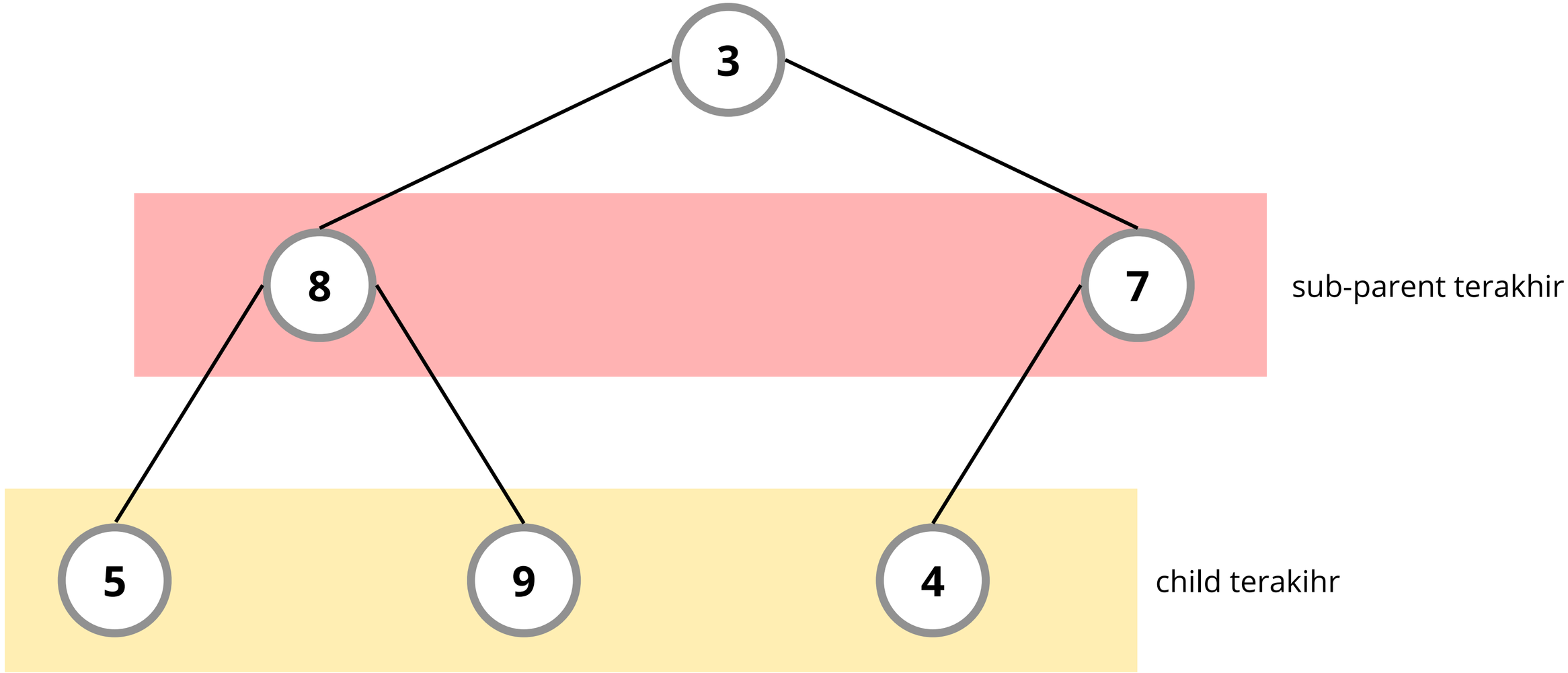
3	8	7	5	9	4

step 1: konversi array menjadi binary tree. Kita akan buat **max-heap** dari ini



3	8	7	5	9	4

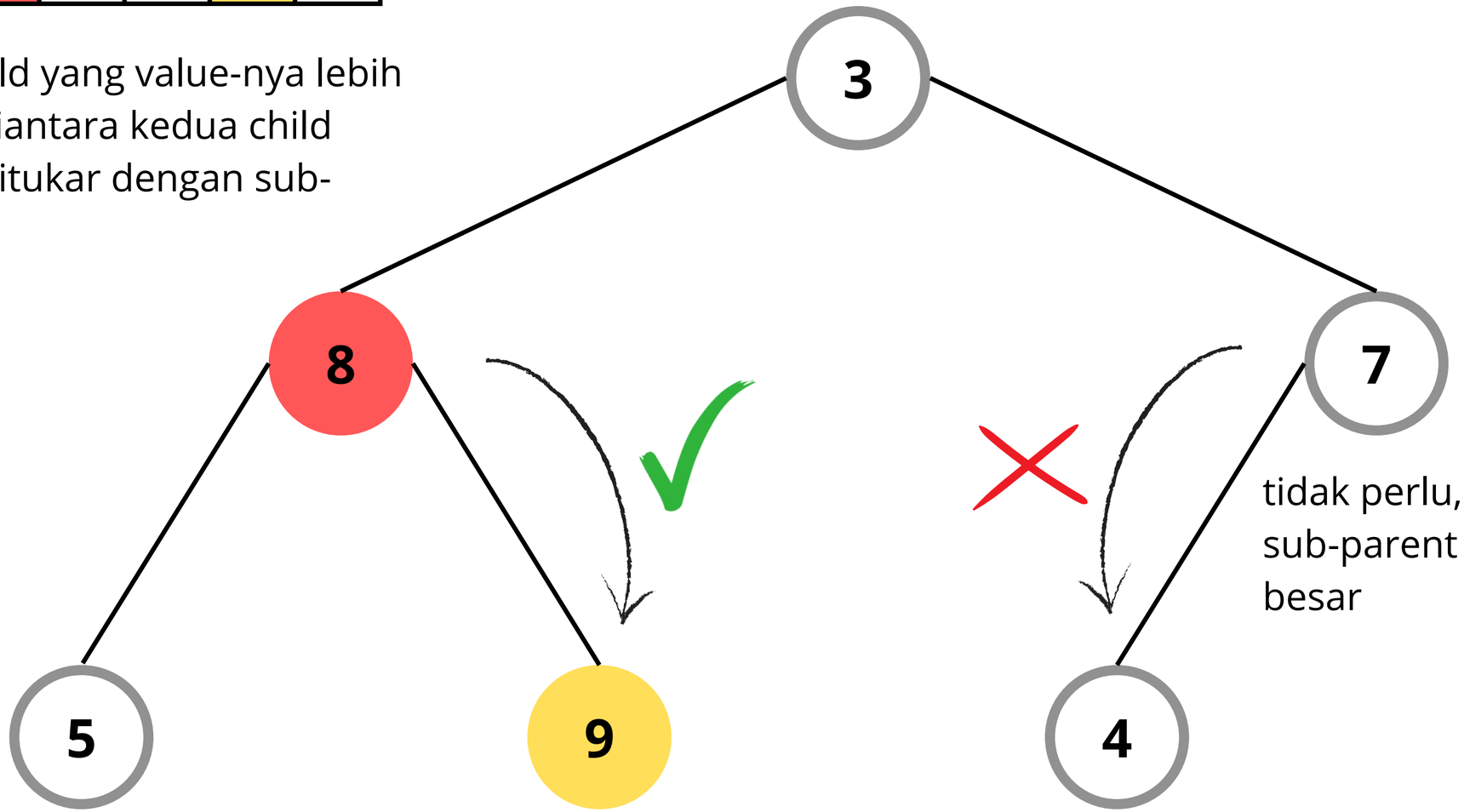
step 2: **Heapify**. identifikasi sub-parent terakhir atau root yang punya child terakhir dahulu



step 2: heapify

3	8	7	5	9	4

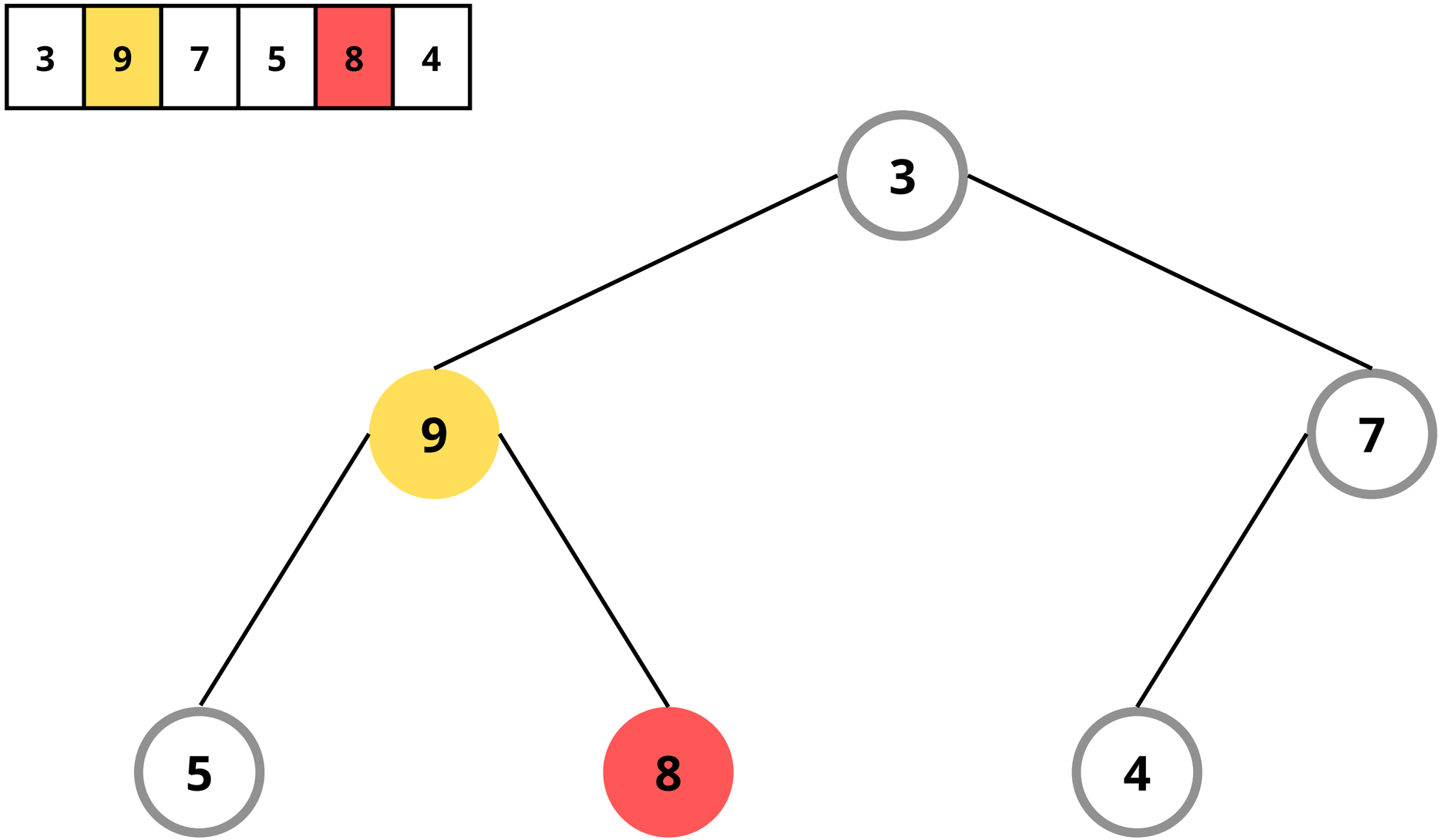
pilih child yang value-nya lebih besar diantara kedua child untuk ditukar dengan sub-parent



tidak perlu, karena value sub-parent sudah lebih besar

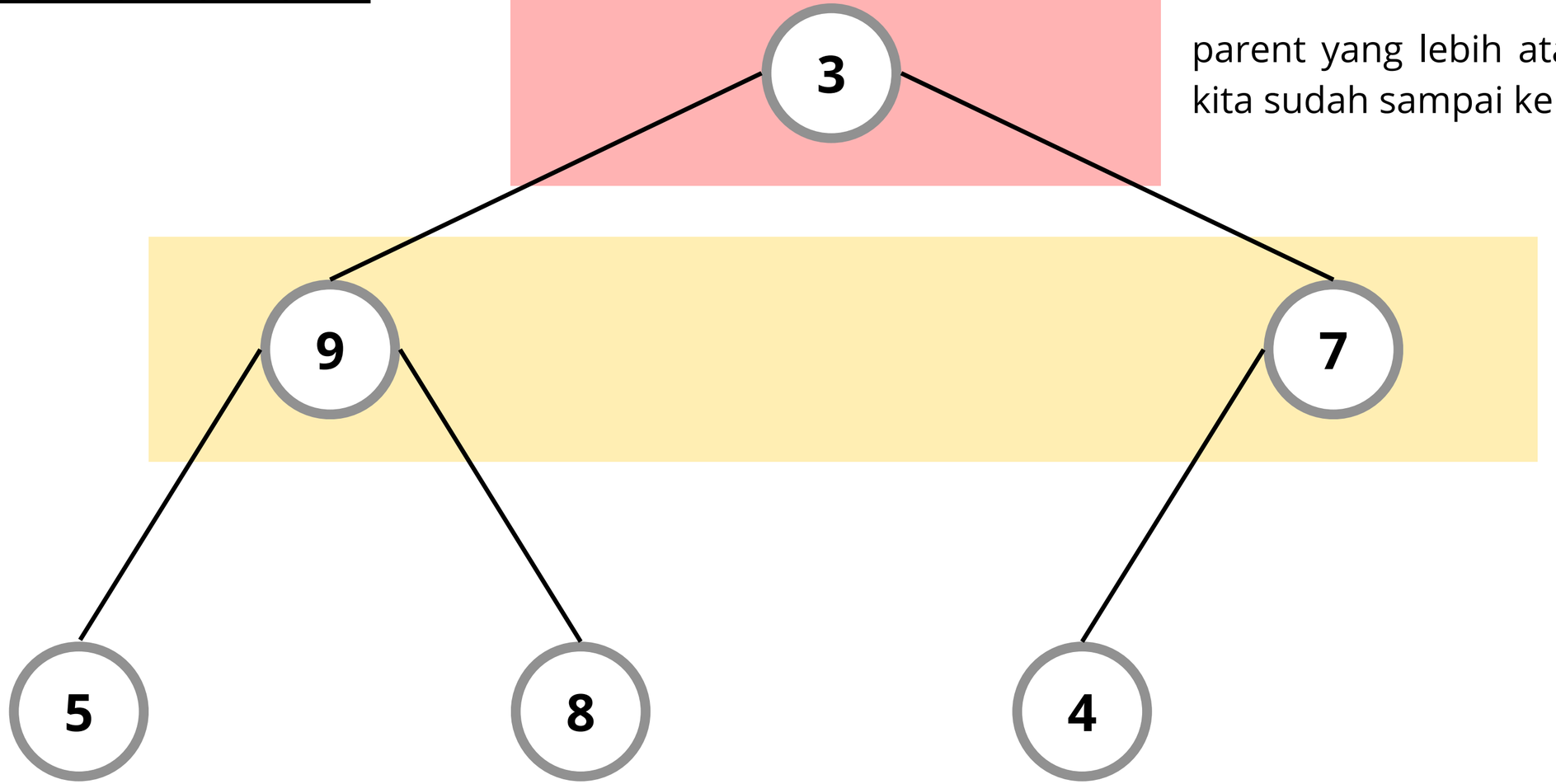
kita fokus ke sub-parent terakhirnya, turun dan tukarkan dengan child yang value-nya lebih besar dari pada sub-parent

step 2: heapify



step 2: heapify

3	8	7	5	9	4



parent yang lebih atas. untuk kasus ini kita sudah sampai ke parent utamnya

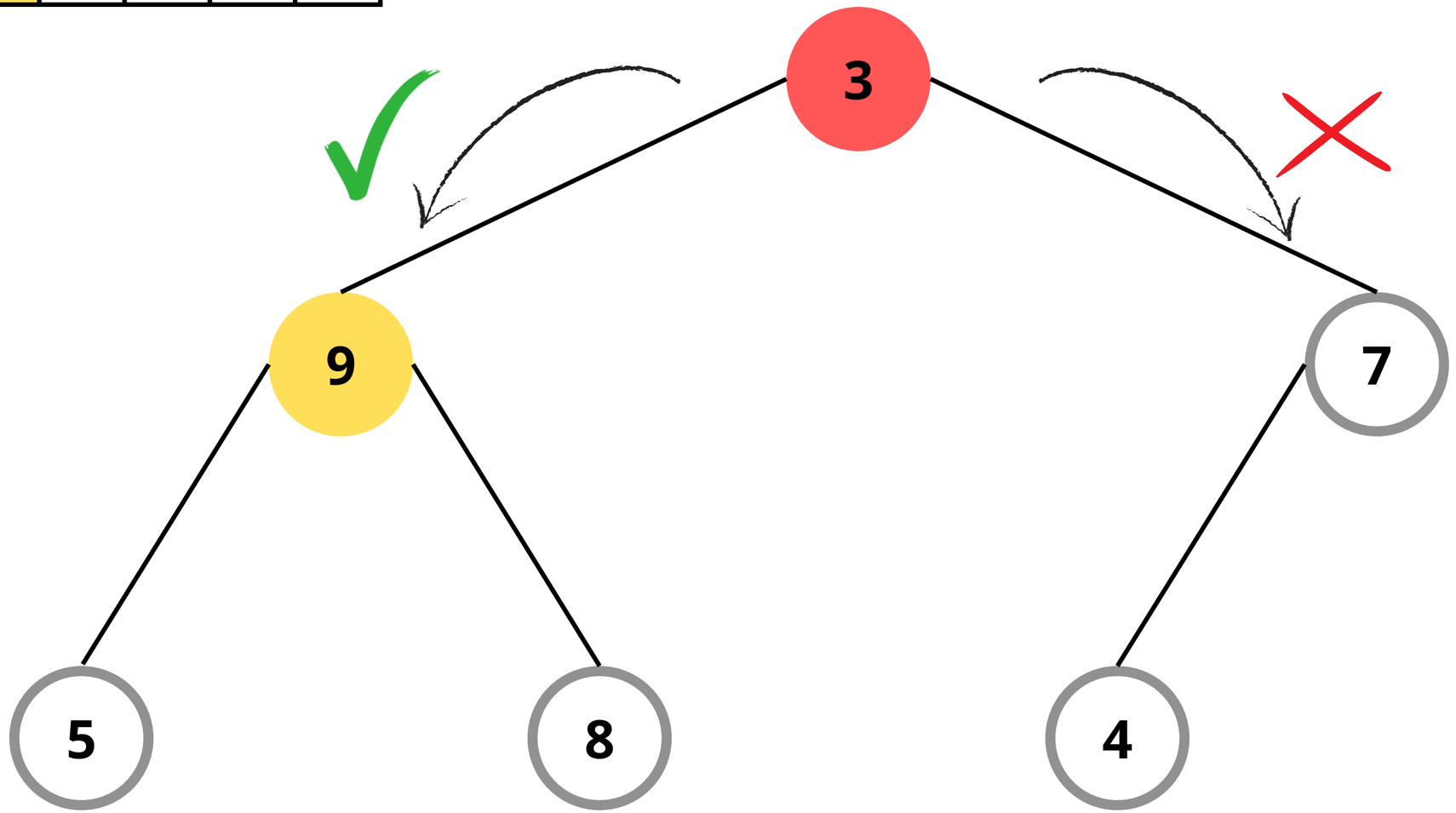
sub-parrent yang akan diheapify

setelah sub-parent terakhir dan child diheapify, kita fokus ke level lebih atas, yaitu root (parent utama) dengan sub-parent

step 2: heapify

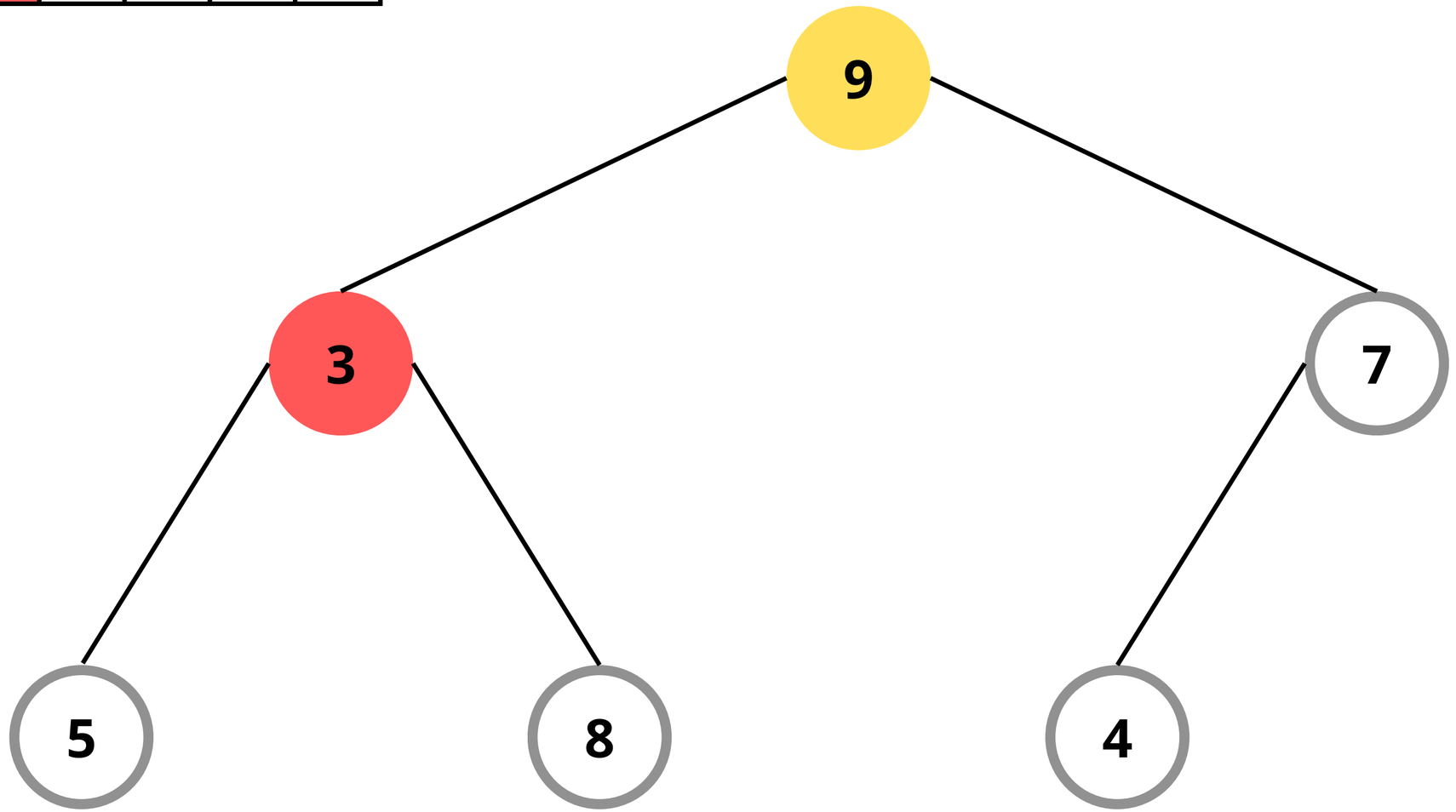
3	9	7	5	9	4

sama seperti konsep sebelumnya, algoritma akan memilih child yang lebih besar value-nya yang akan digantikan dengan parent di atasnya

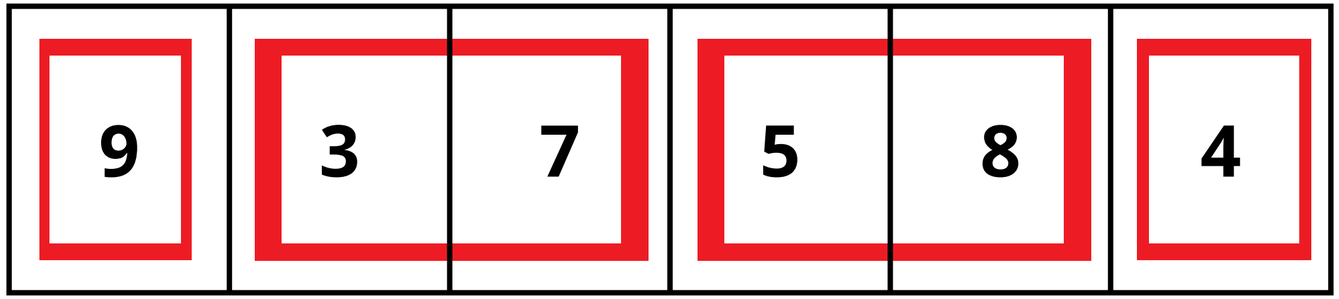


step 2: heapify

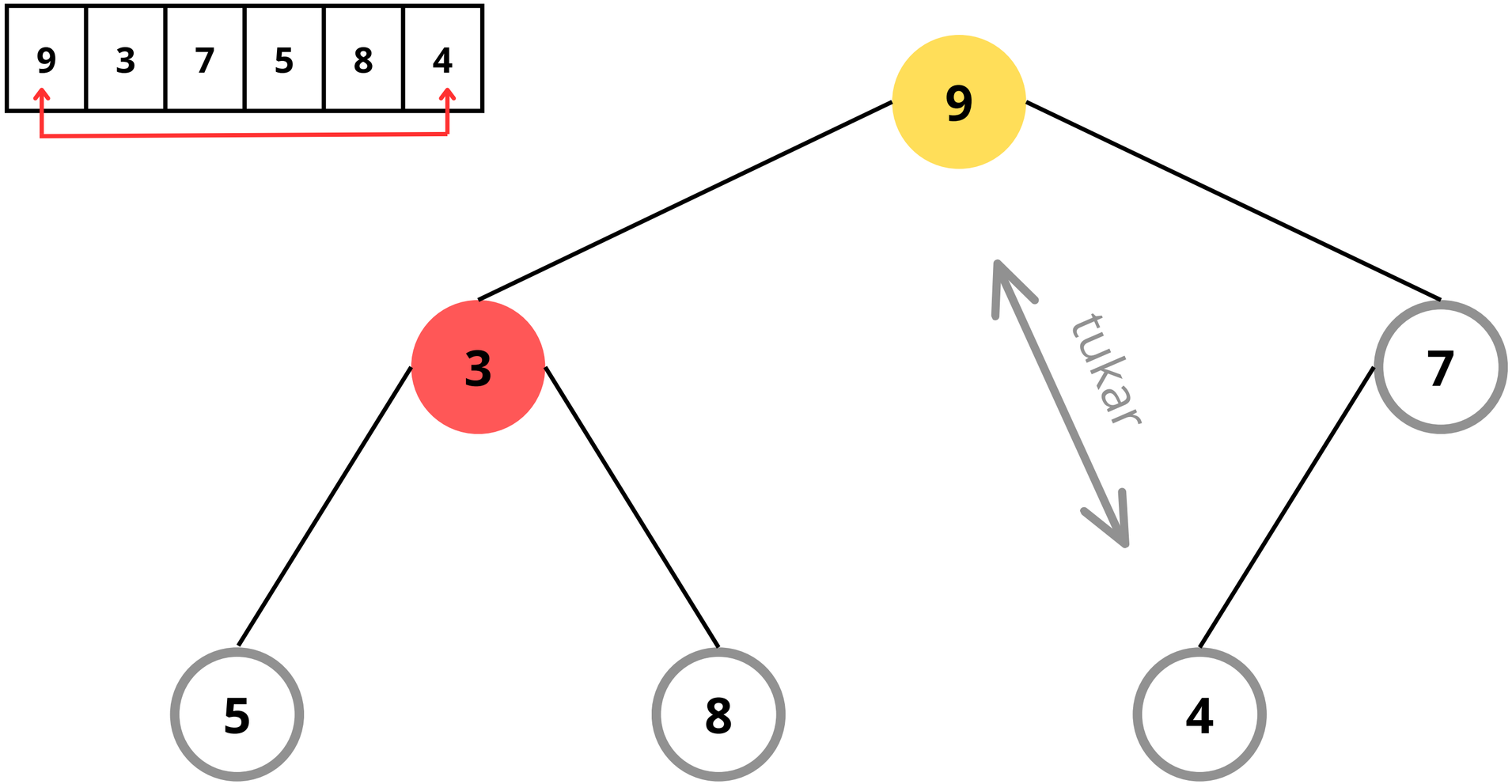
9	3	7	5	9	4



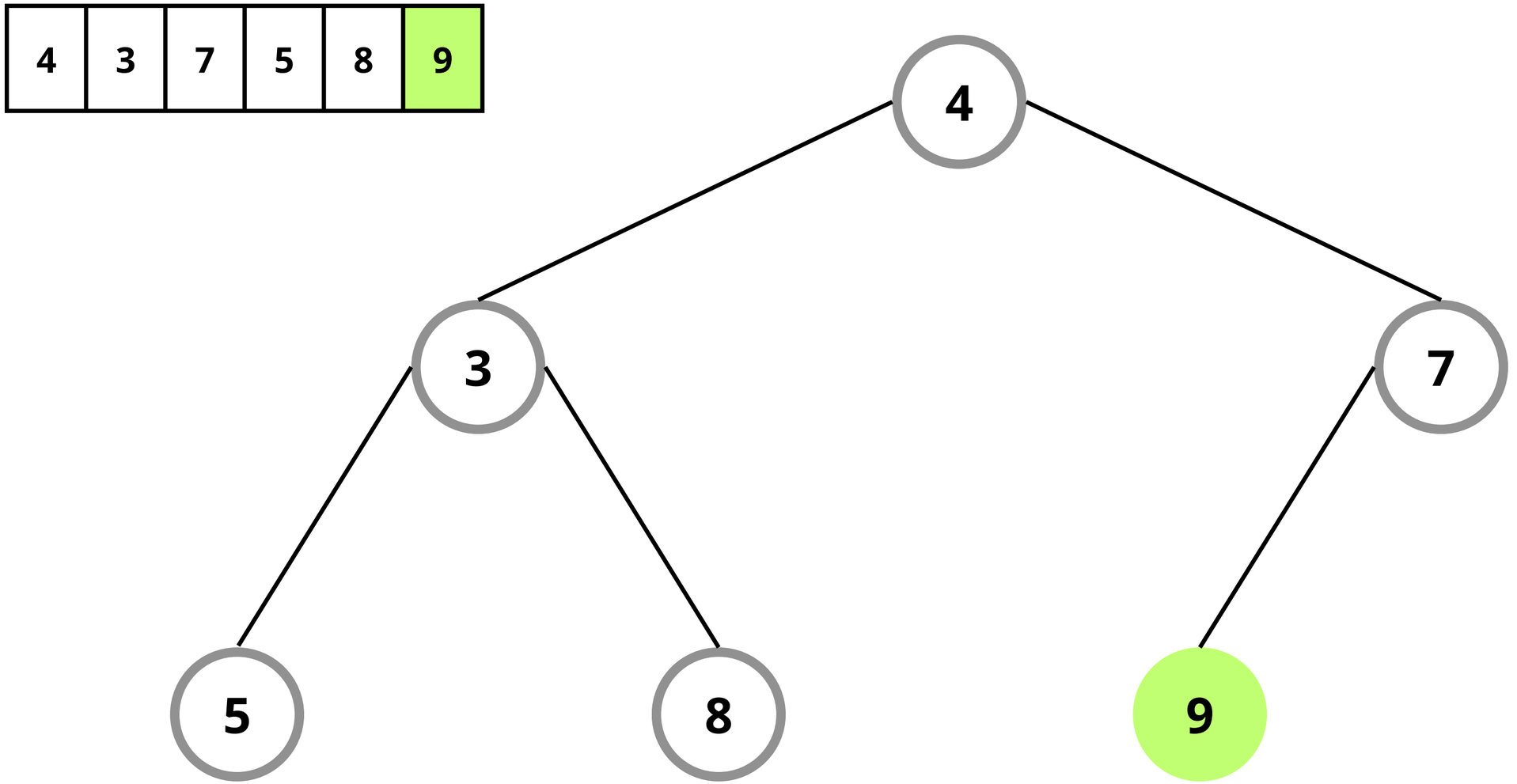
dari heapify yang didapat kita bentuk kembali array nya



step 3: tukar elemen pertama/root dengan elemen array terakhir/child terakhir heap

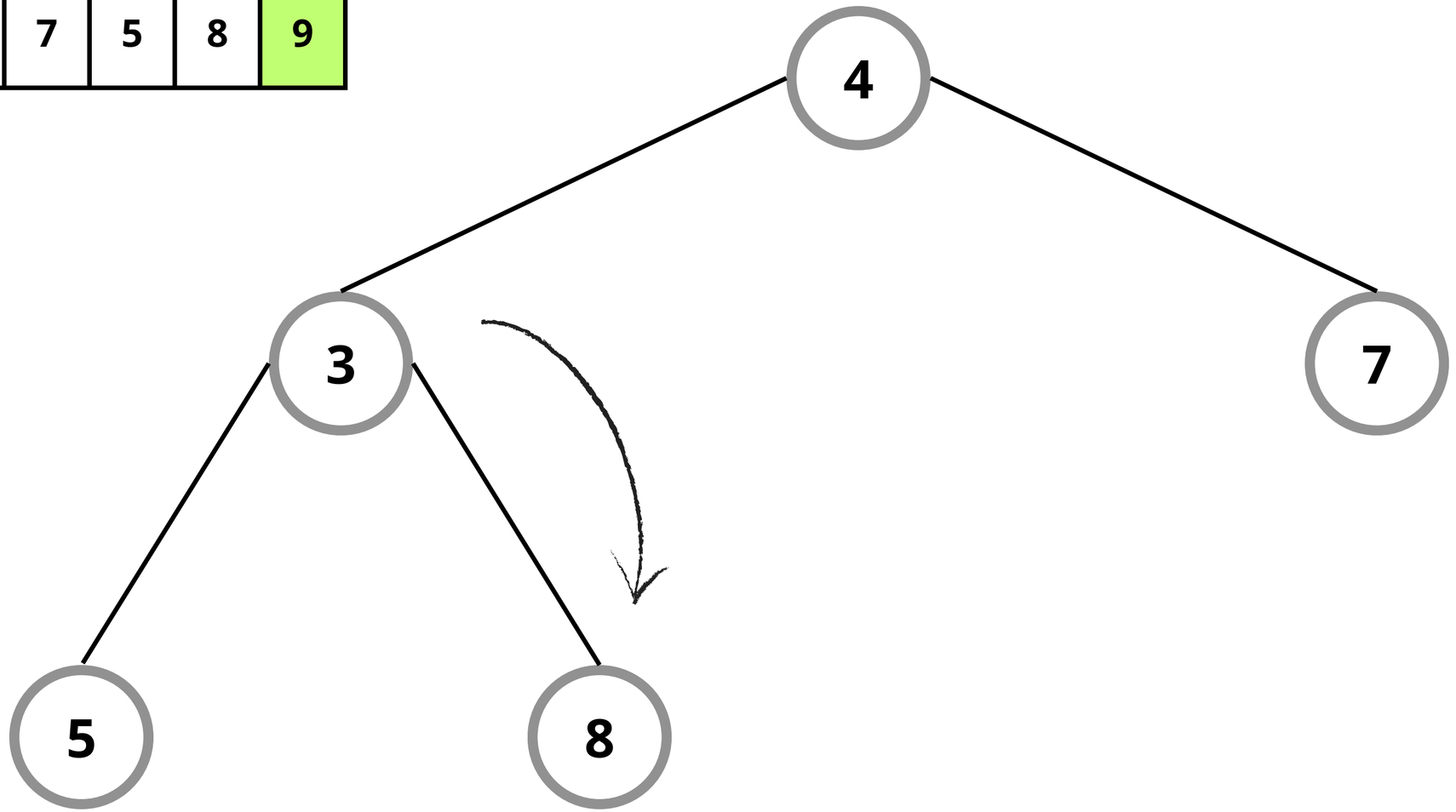


elemen terakhir telah ditetapkan dan tidak akan diproses lebih lanjut lagi

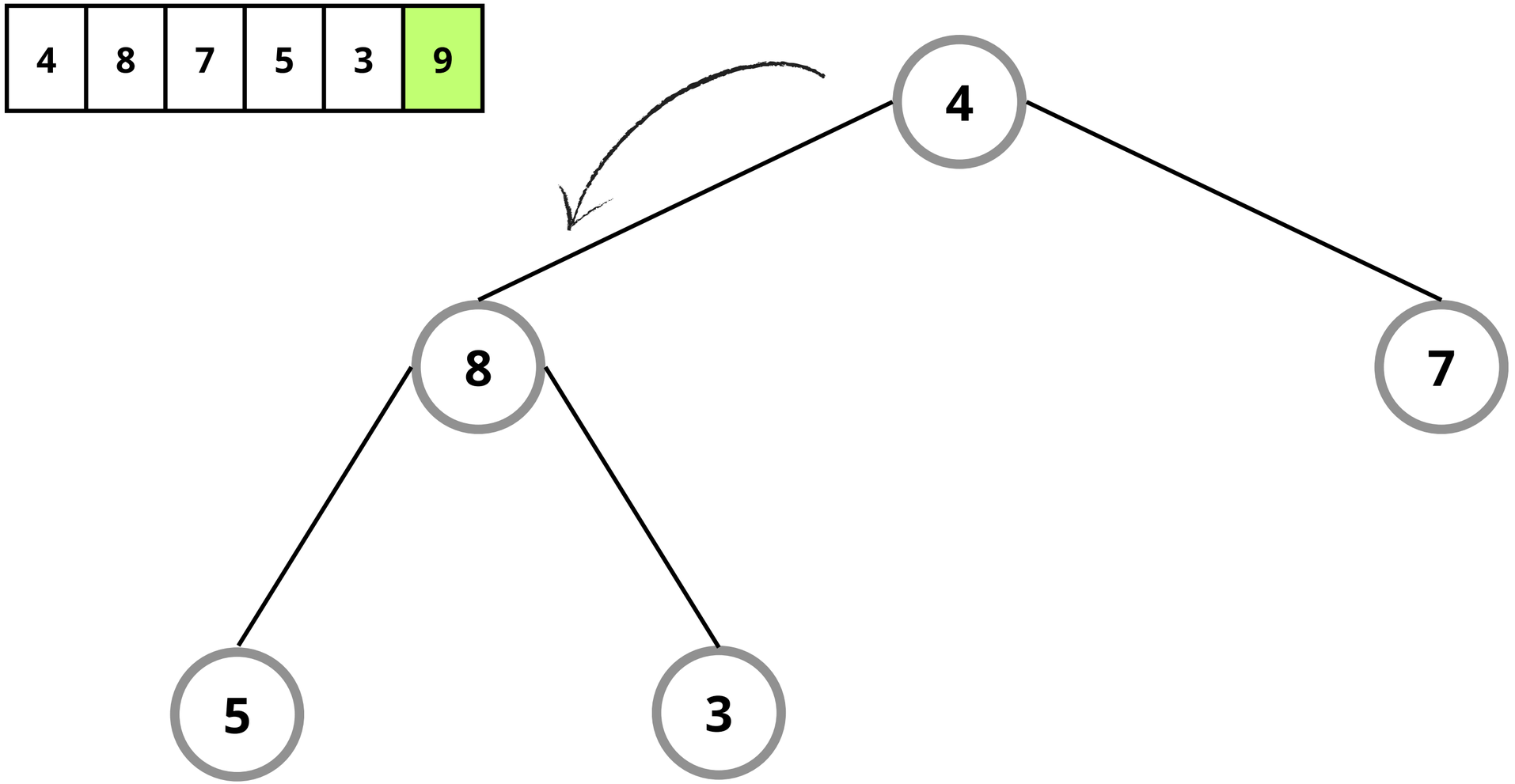


step 4: ulangi proses heapify seperti sebelumnya hingga terbentuk max-heap dan array disort sesuai urutan

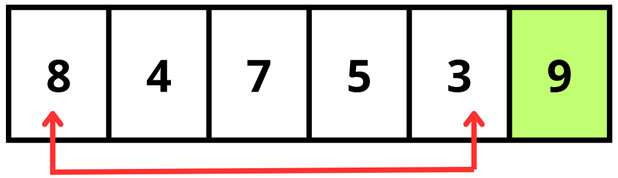
4	3	7	5	8	9



Heapify

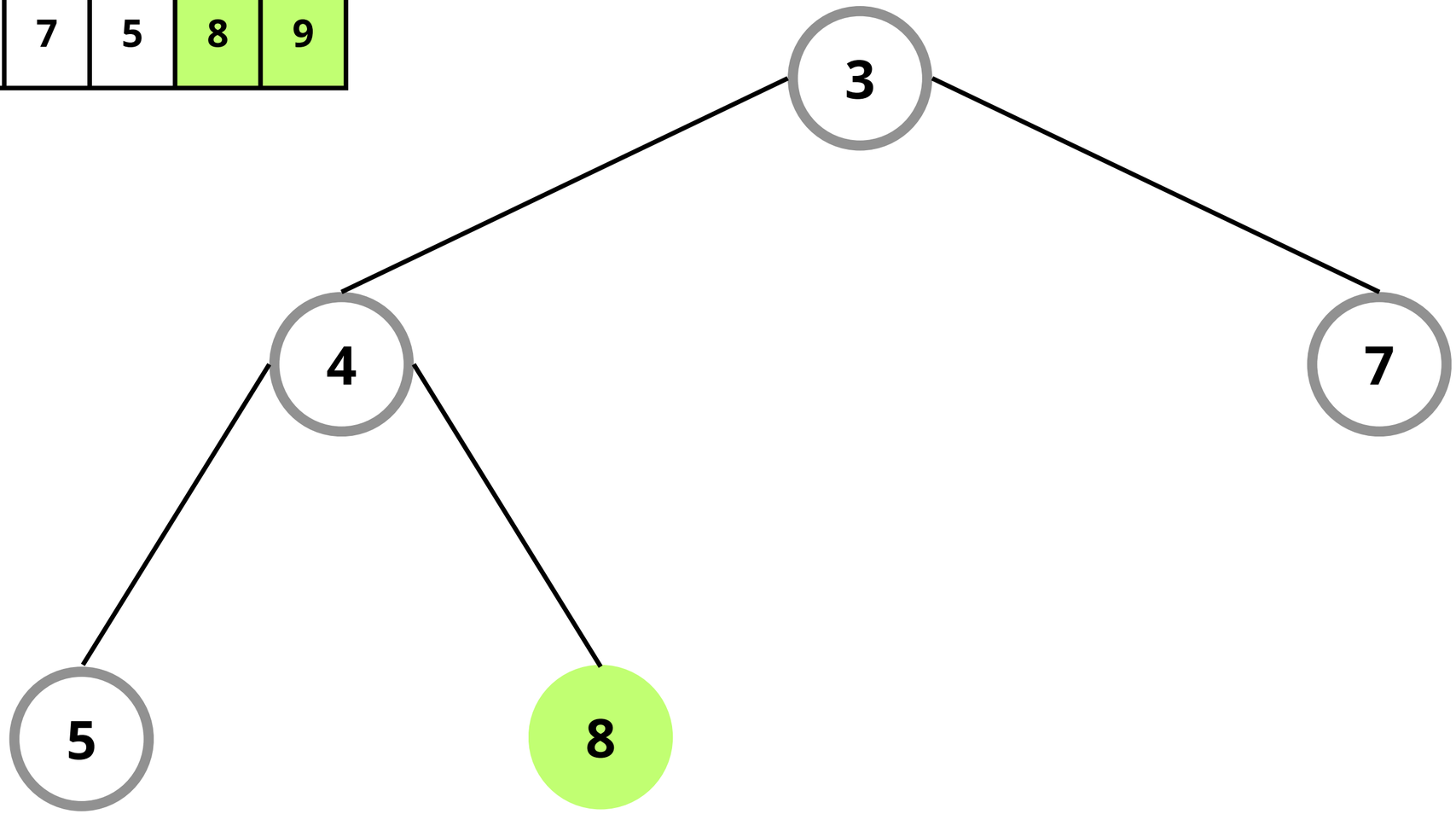


Tukar root dengan child elemen terakhir



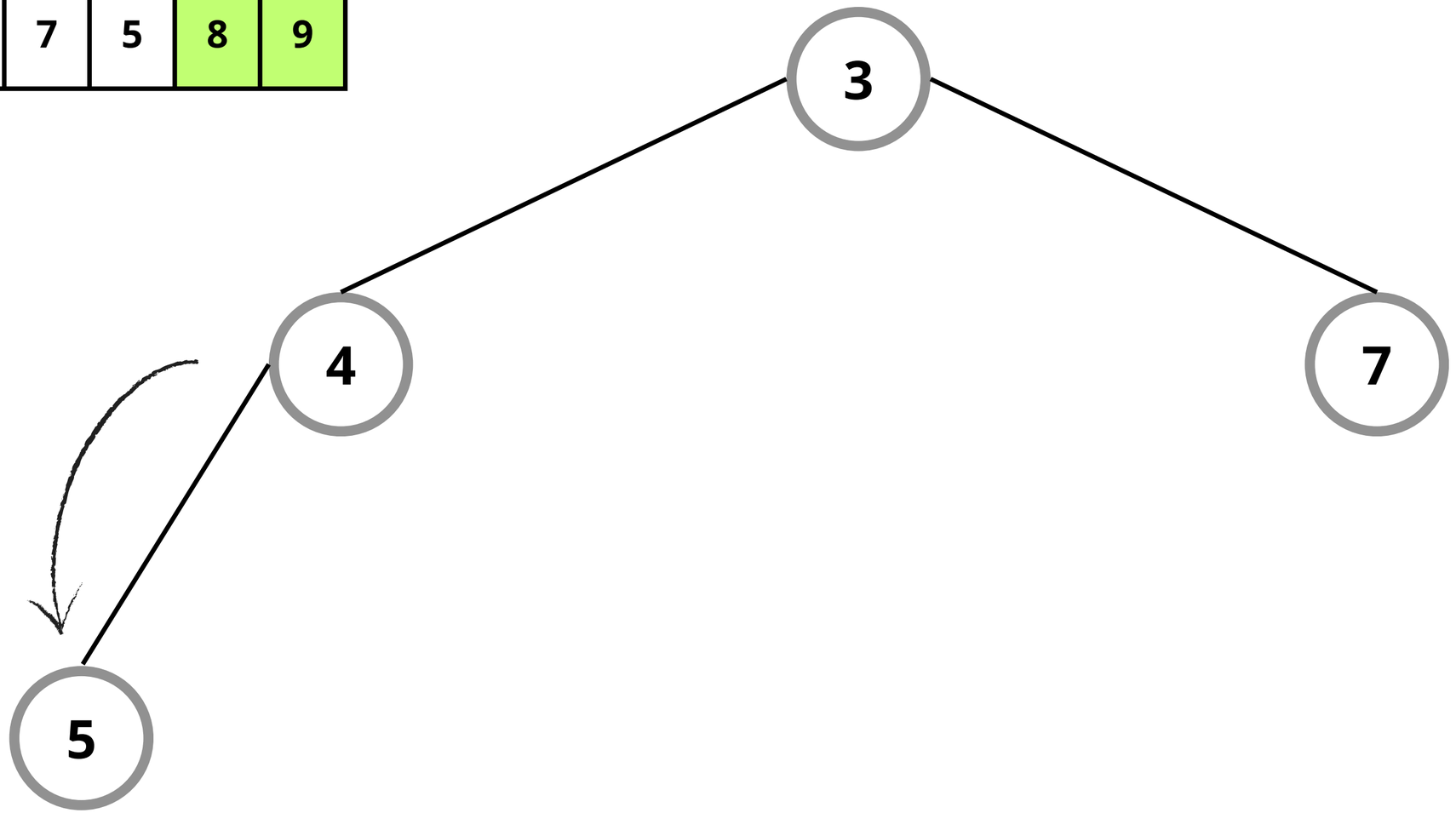
Tukar root dengan child elemen terakhir

3	4	7	5	8	9



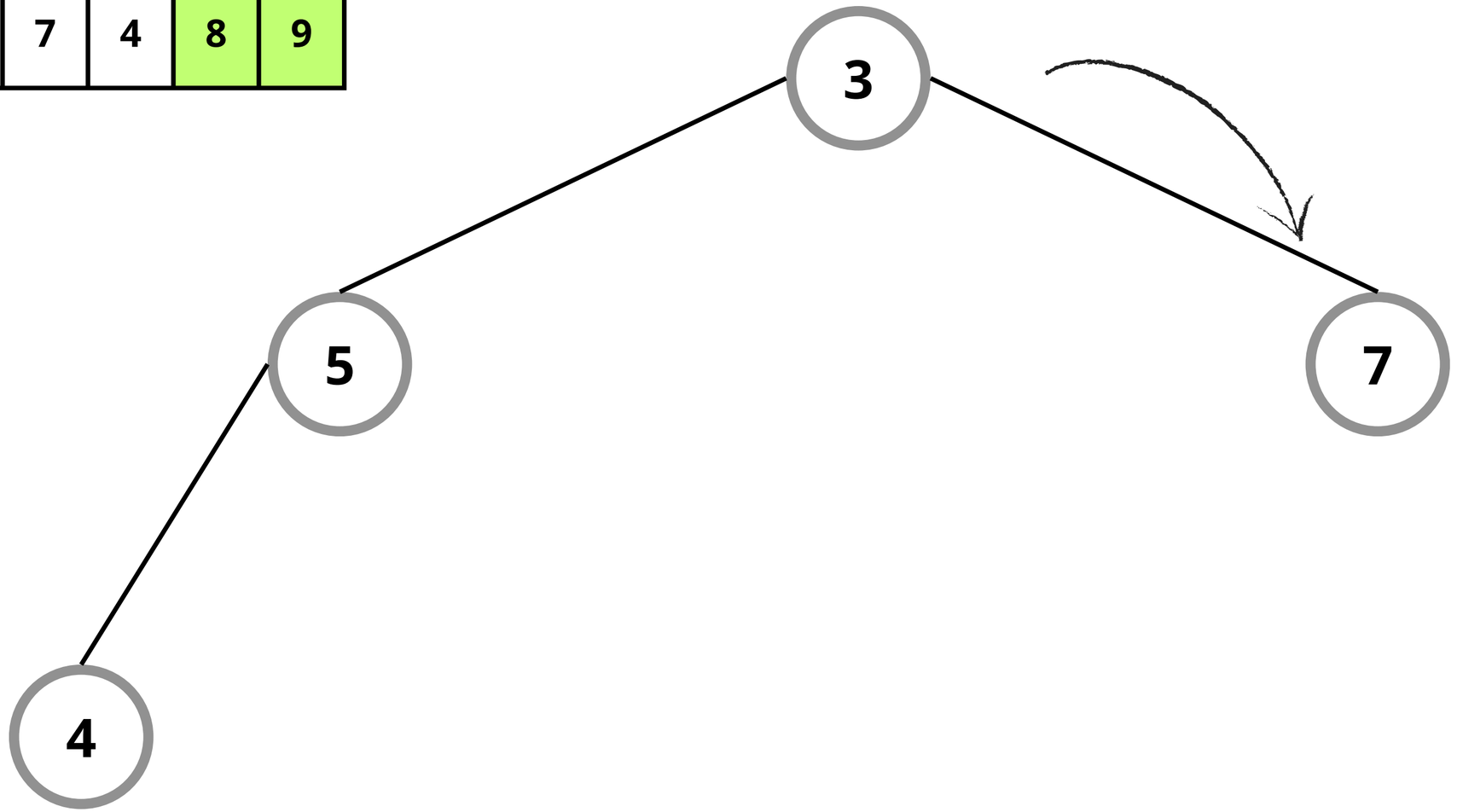
ulangi heapify seterusnya sampai terbentuk max-heap

3	4	7	5	8	9

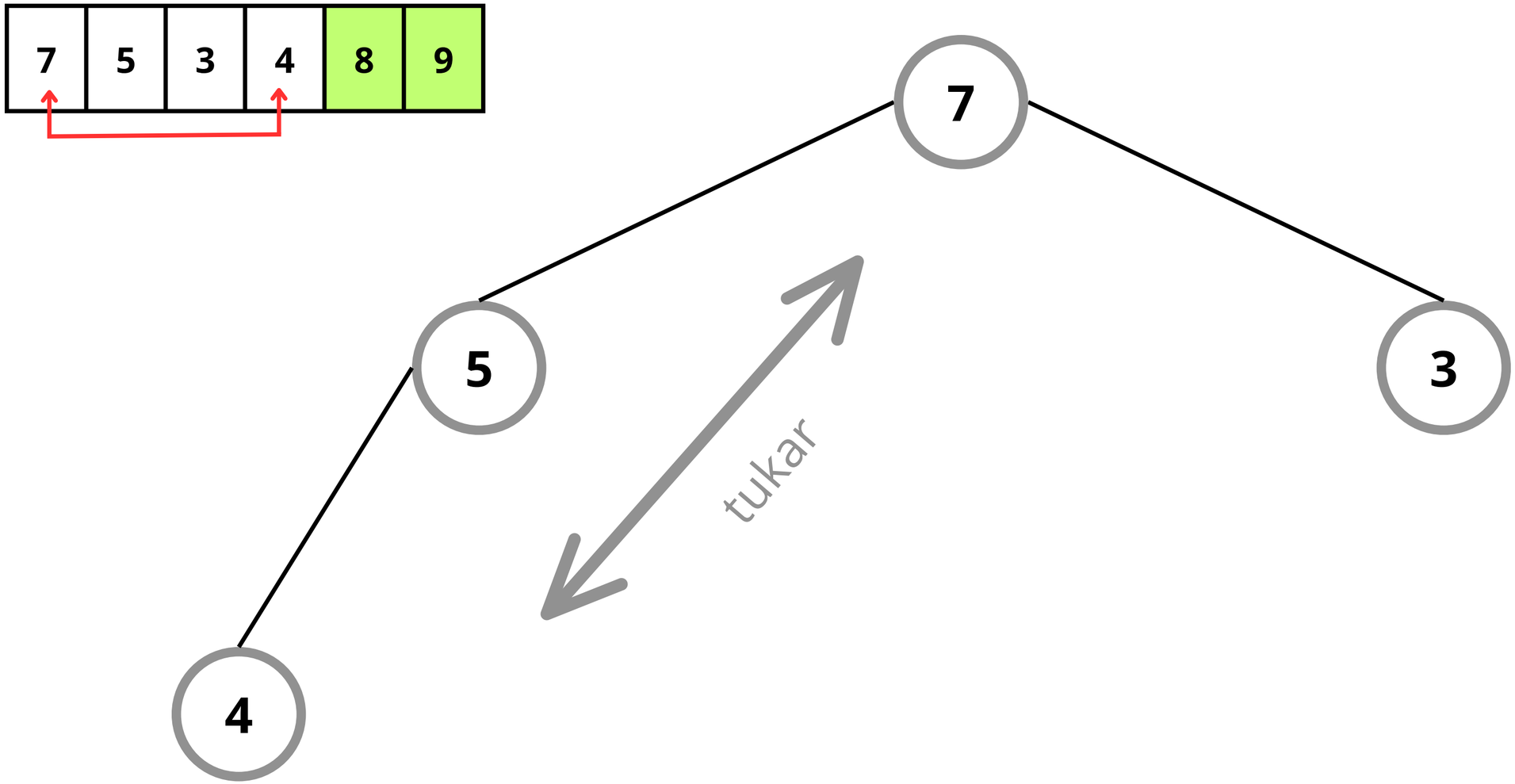


ulangi heapify seterusnya sampai terbentuk max-heap

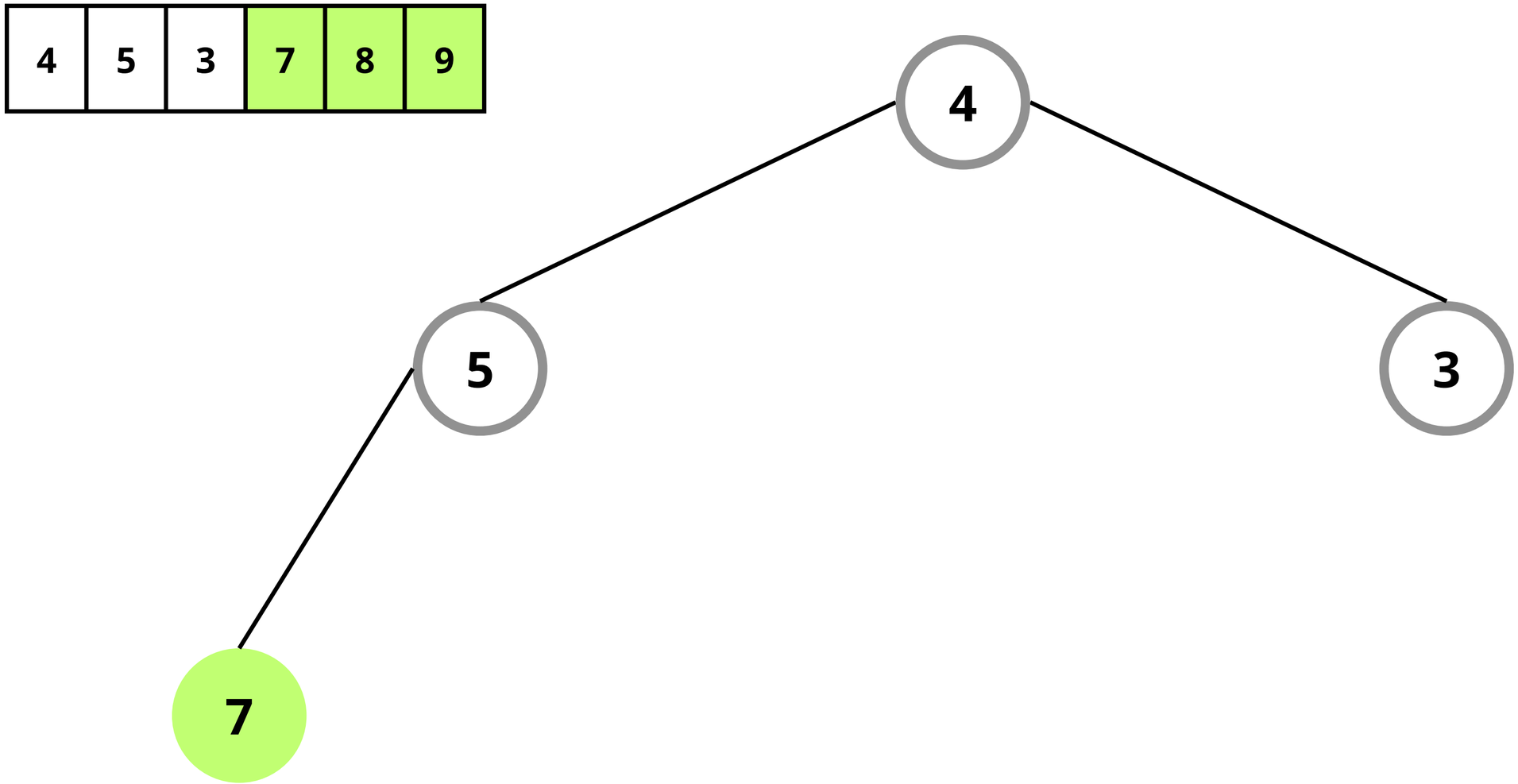
3	5	7	4	8	9



tukar root dengan child elemen terakhir

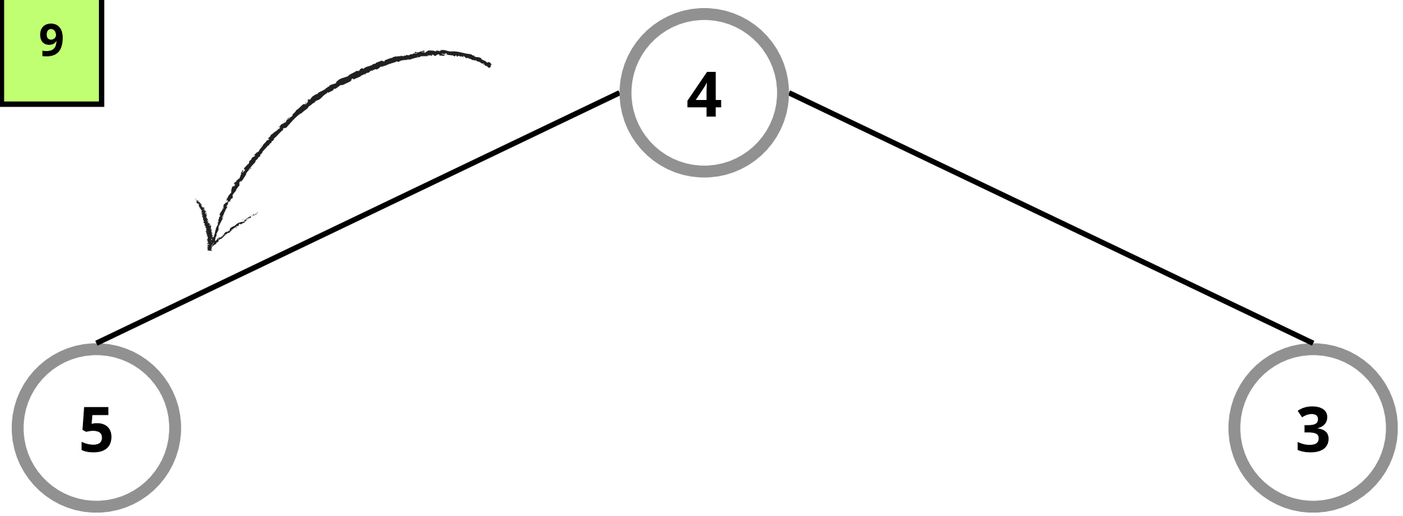


tukar root dengan child elemen terakhir

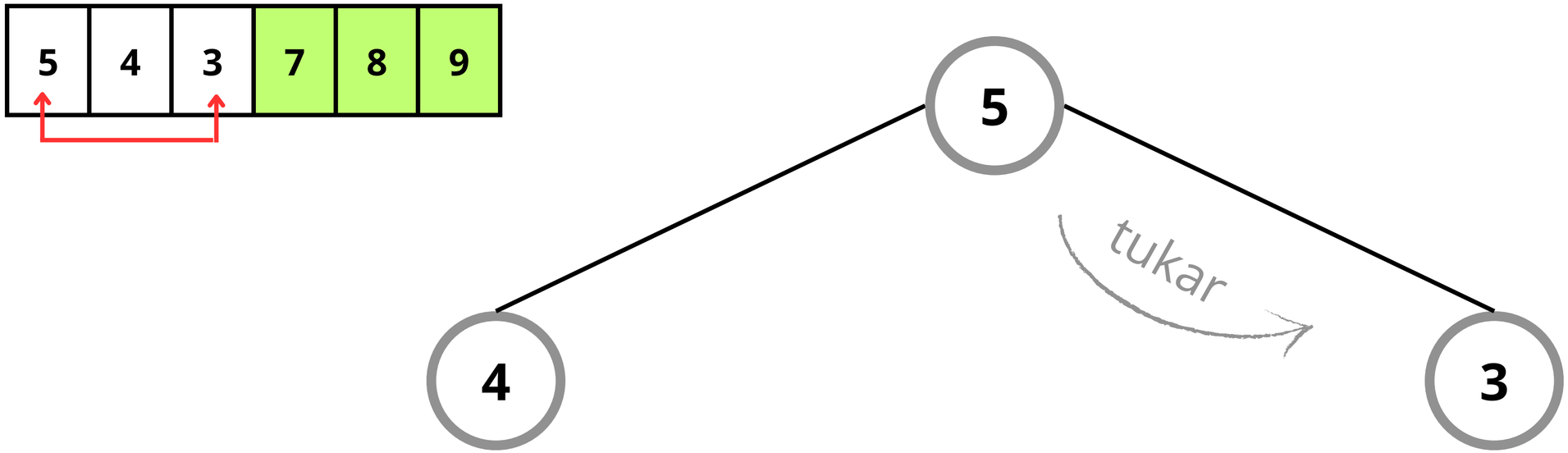


ulangi heapify seterusnya sampai terbentuk max-heap

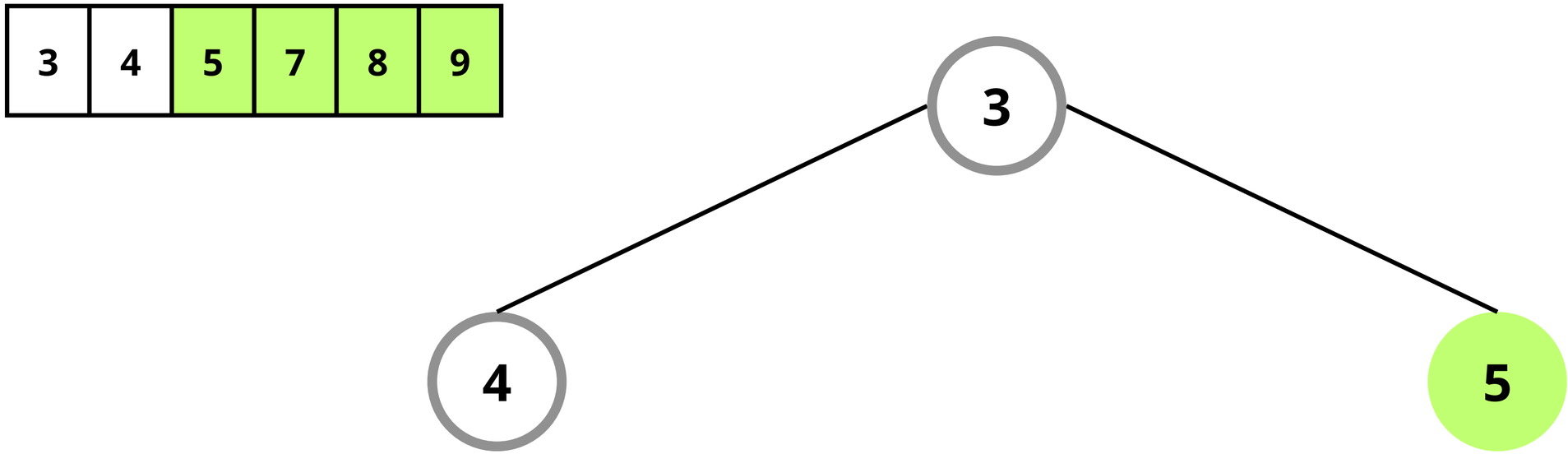
4	5	3	7	8	9



tukar root dengan child elemen terakhir

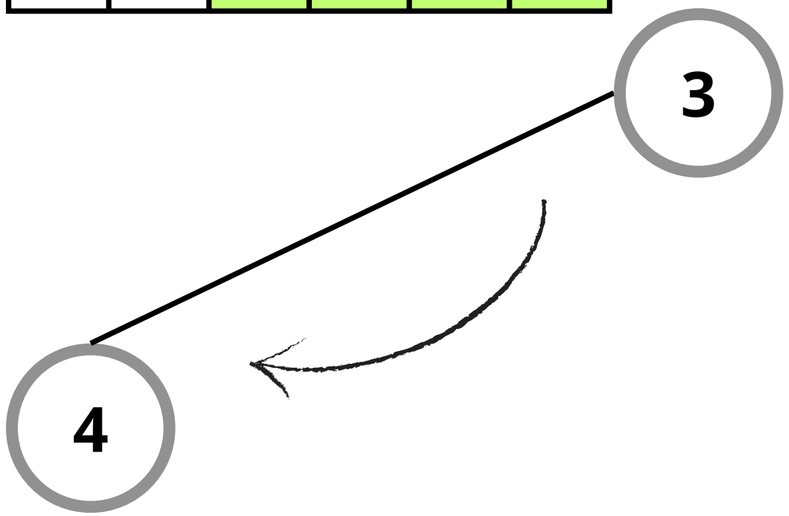


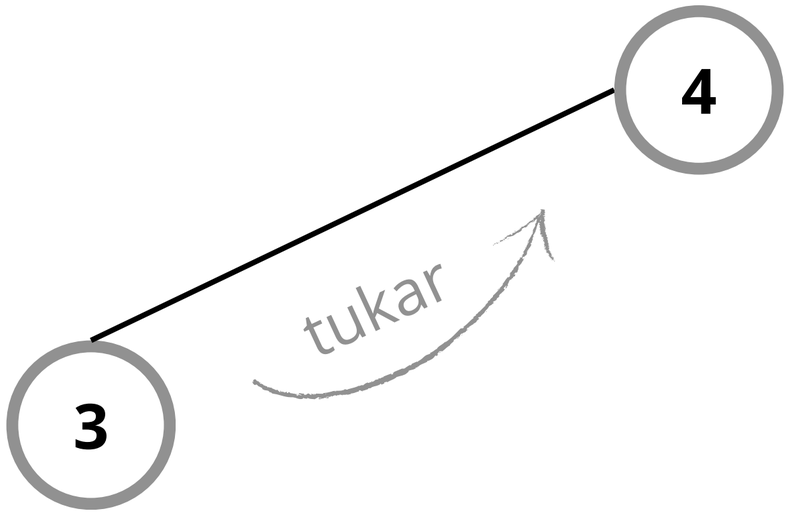
tukar root dengan child elemen terakhir

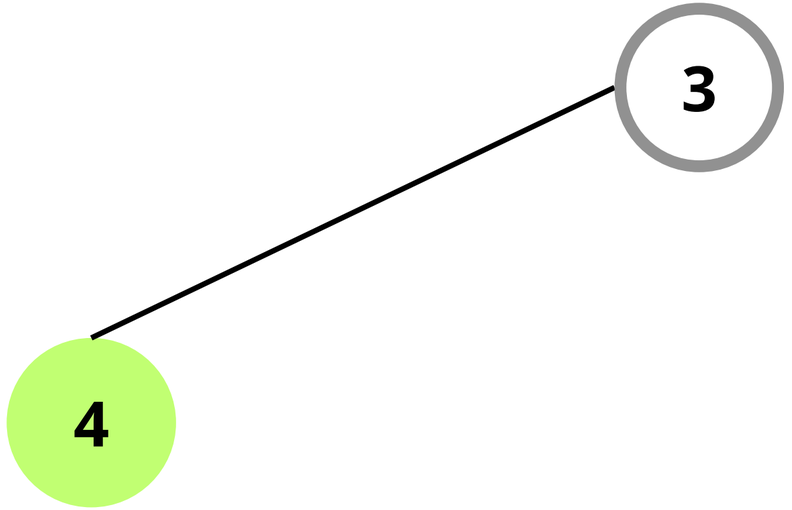


ulangi heapify seterusnya sampai terbentuk max-heap

3	4	5	7	8	9







tukar root dengan child elemen terakhir

3	4	5	7	8	9