

# ПОСТРОЕНИЕ СБАЛАНСИРОВАННОГО ДЕРЕВА ПОИСКА

*Идеальное сбалансированное дерево поиска* – это двоичное дерево, в котором **число вершин** в левых и правых поддеревьях отличается не более чем на 1.

*Сбалансированное дерево поиска* – это двоичное дерево, в котором **высоты** левых и правых поддеревьев каждой из его вершин отличаются не более чем на 1.

# АЛГОРИТМ ПОСТРОЕНИЯ СБАЛАНСИРОВАННОГО ДЕРЕВА ПОИСКА

При добавлении узла считаем баланс его «отца» (p) и «деда» (gp) и всех остальных «предков»

Баланс узла определяется как разность высот его правого и левого поддеревя:

$$h = R - L$$

Если для какого-то узла (u)  $h(gp) = 2$  или  $-2$  то делаем однократный или двукратный поворот.

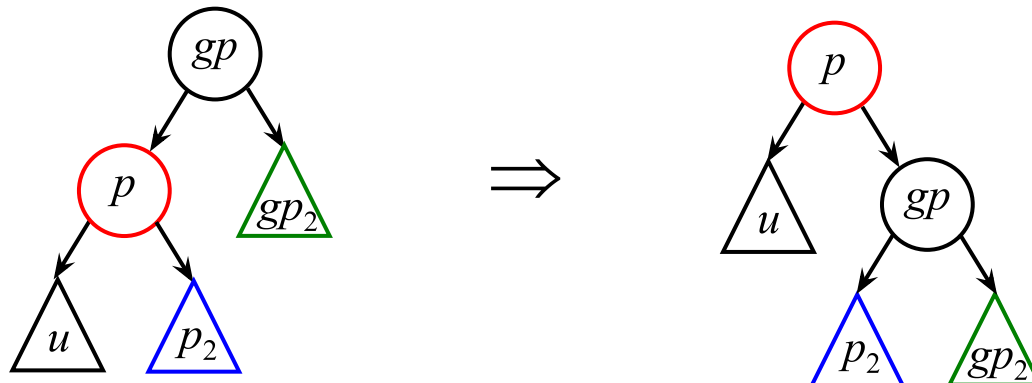
А именно:

а) если  $h(gp) \cdot h(p) > 0$  и  $h(p) < 0$  — R  
(один раз поворот направо относительно p)

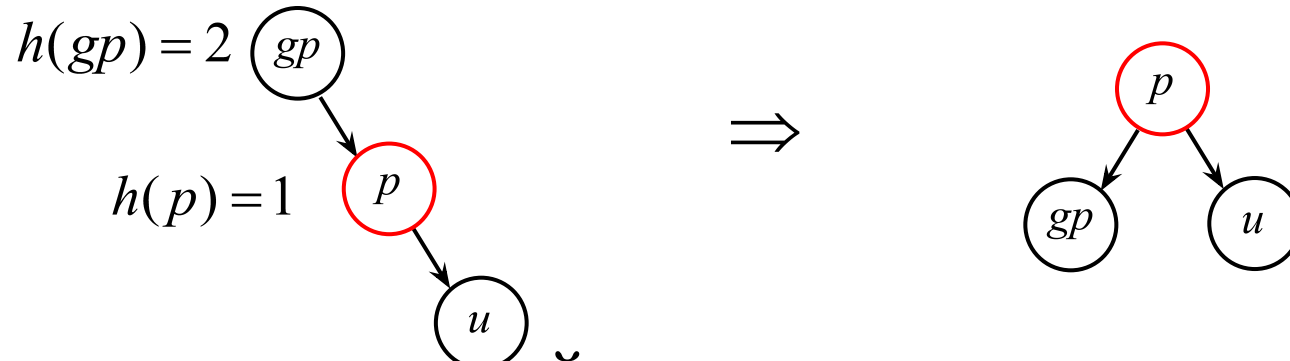


(т.е. p станет «главой семьи», gp — правым «сыном»)

Более сложная ситуация:

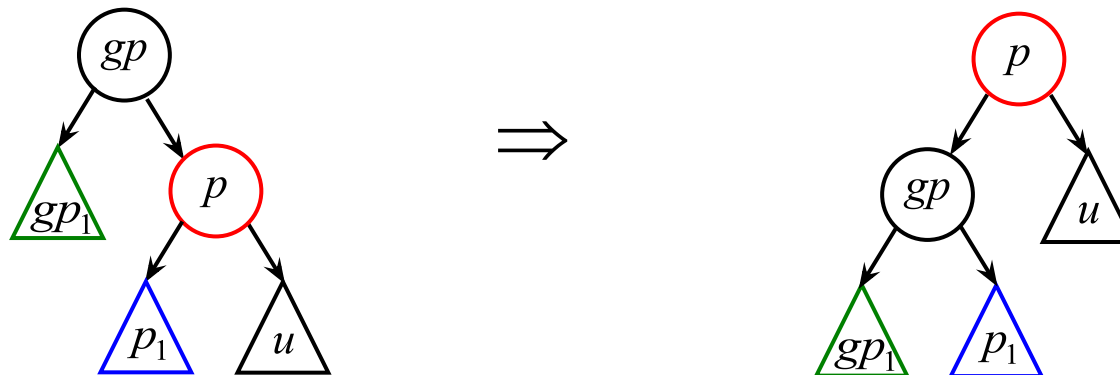


б) если  $h(gp) \cdot h(p) > 0$  и  $h(p) > 0$  — L  
 (один раз поворот налево относительно p)



(т.е.  $p$  станет «главой семьи»,  $gp$  — левым «сыном»)

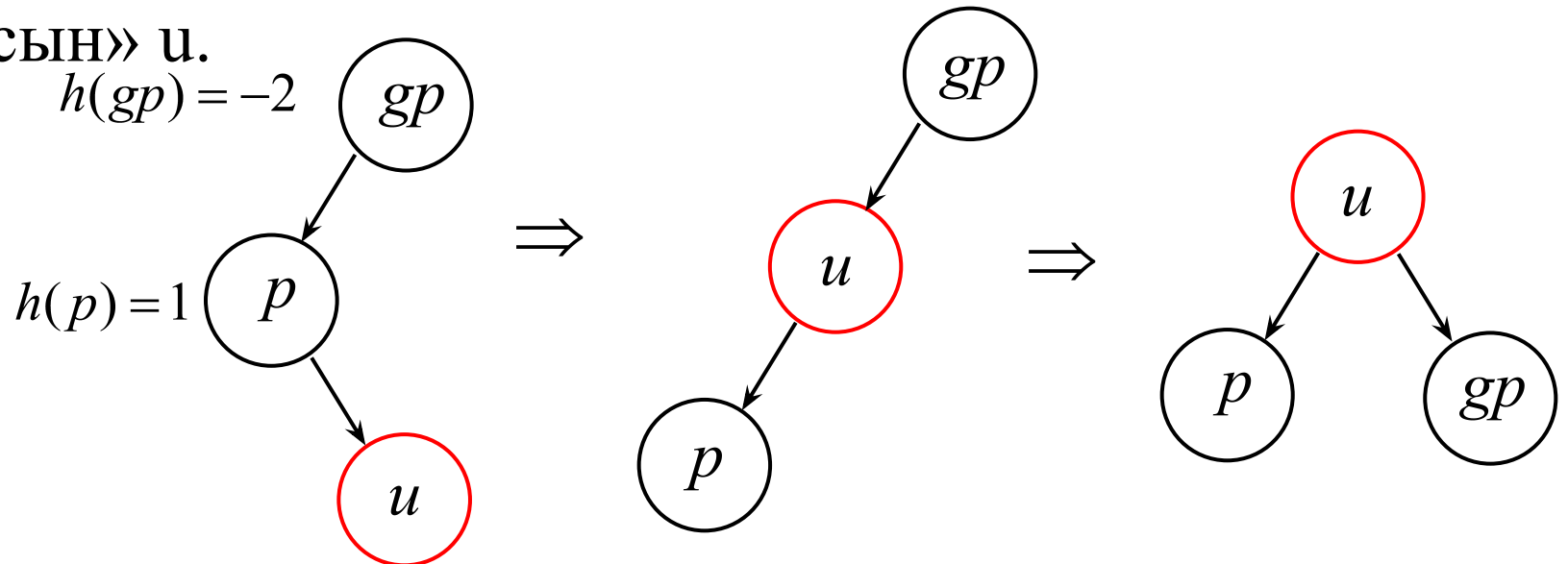
Более сложная ситуация:



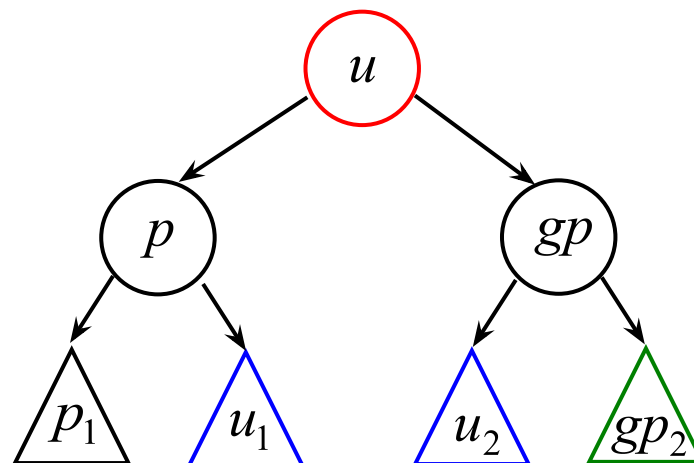
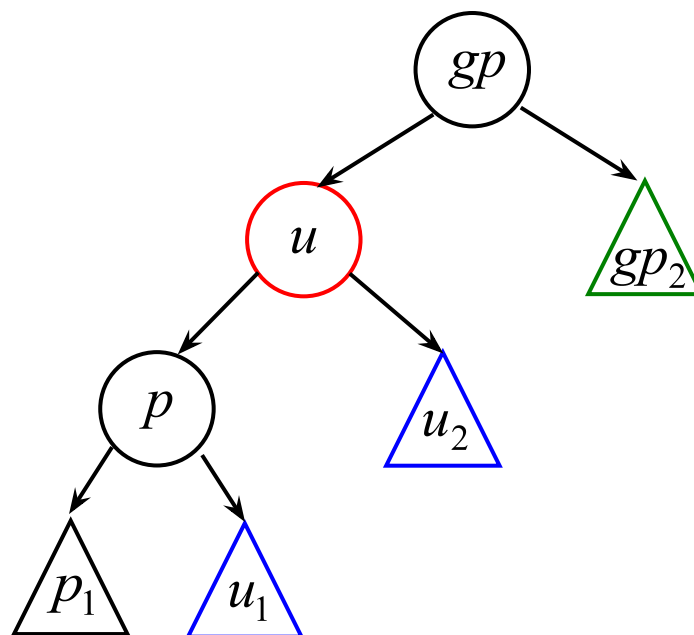
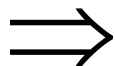
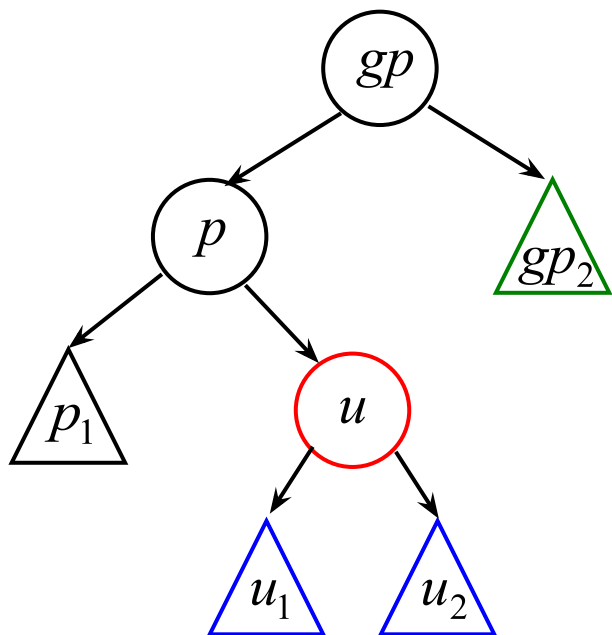
в) если  $h(gp) \cdot h(p) < 0$  и  $h(p) > 0$  – двукратный поворот LR

- т.е. 1) сначала налево относительно  $u$   
(«отец» и «сын» меняются ролями),  
2) затем направо относительно  $u$   
(«сын» становится «главой семьи»)

В итоге:  $p$  станет левым «сыном»,  $gp$  – правым «сыном», «родителем» станет бывший правый «сын»  $u$ .



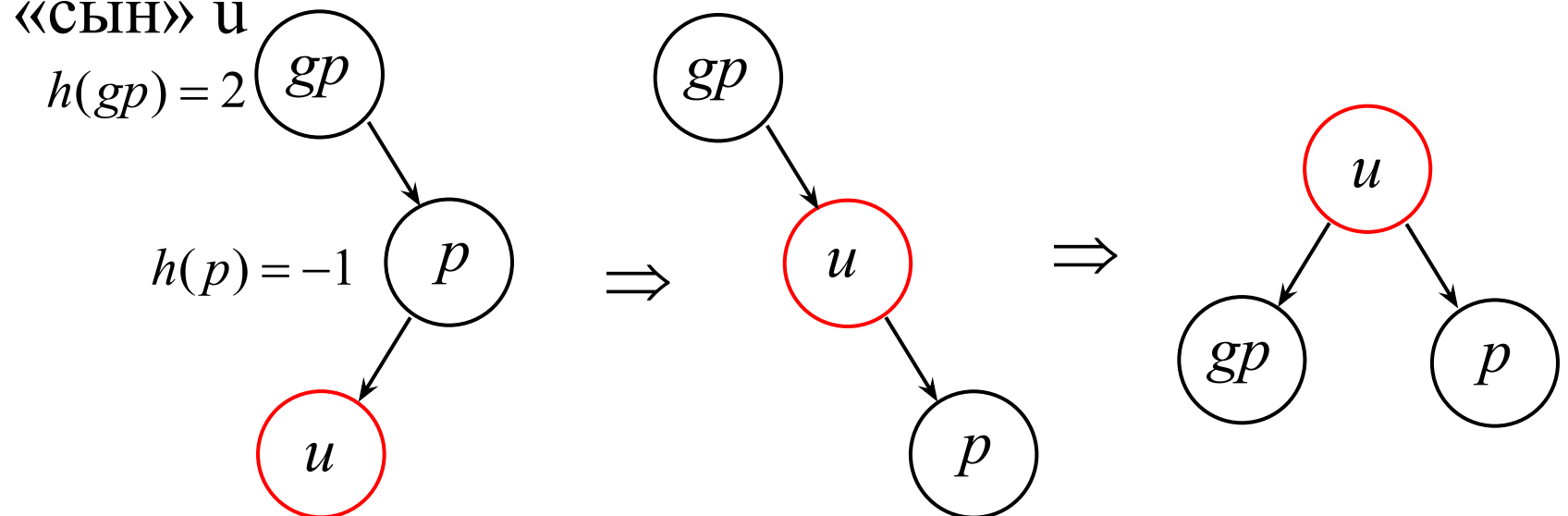
Более сложная ситуация поворота LR:



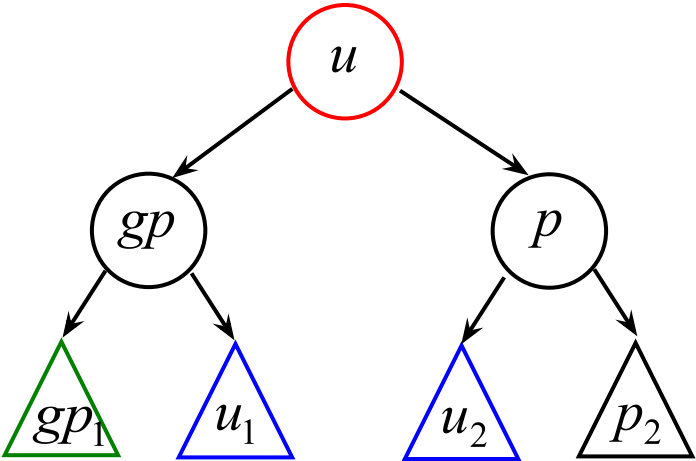
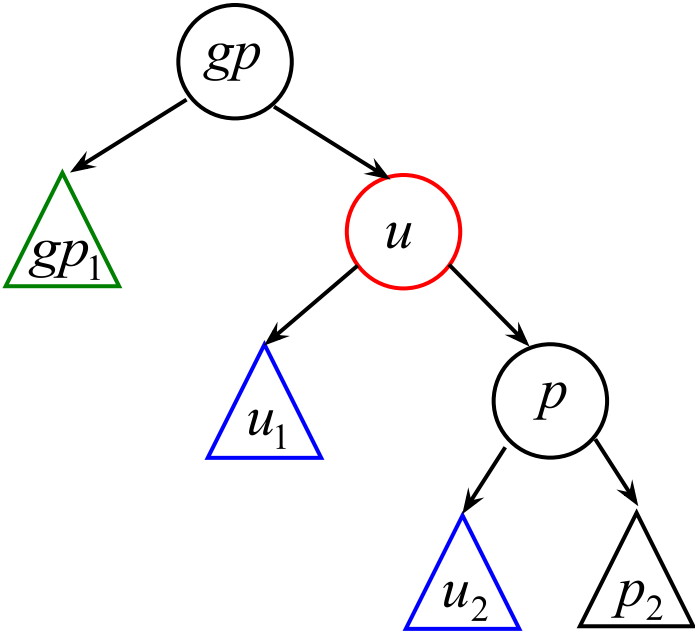
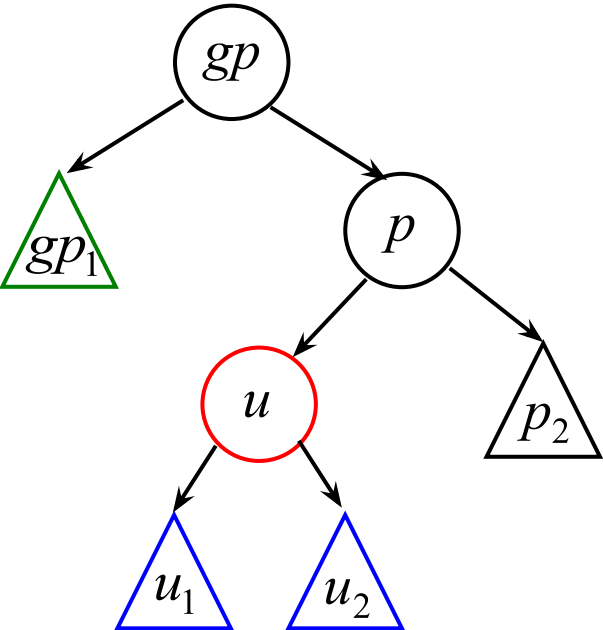
г) если  $h(gp) \cdot h(p) < 0$  и  $h(p) < 0$  – двукратный поворот RL

- т.е. 1) сначала направо относительно  $u$   
(«отец» и «сын» меняются ролями),  
2) затем налево относительно  $u$   
(«сын» становится «главой семьи»).

В итоге:  $p$  станет правым «сыном»,  $gp$  – левым «сыном», «родителем» станет бывший левый «сын»  $u$



Более сложная ситуация поворота RL:

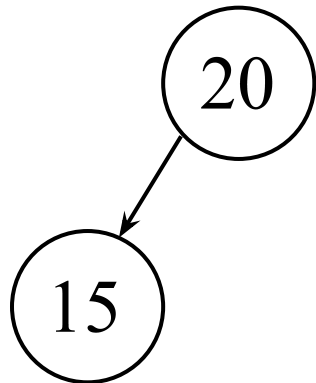




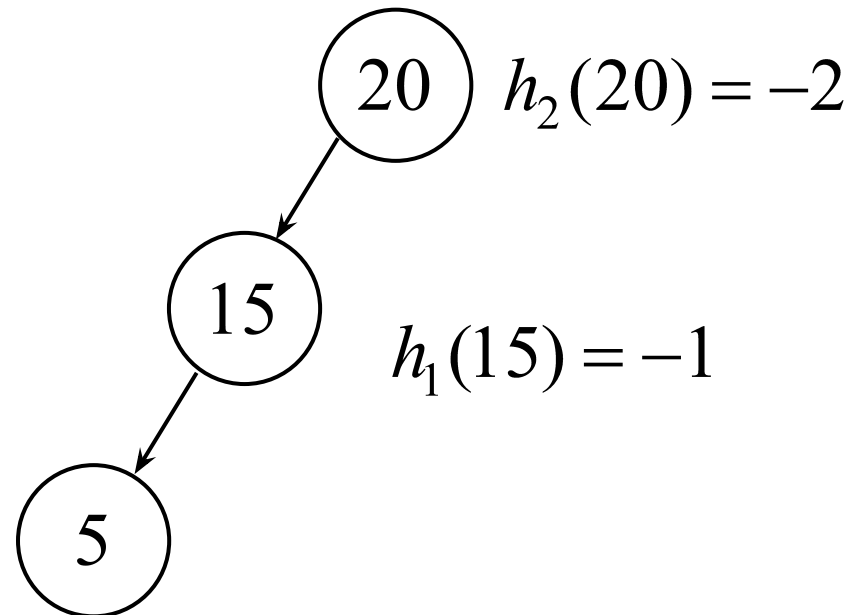
Задача. Построить сбалансированное дерево для массива ключей

{20, 15, 5, 30, 55, 25, 10, 6, 2, 17, 35, 40, 27, 11, 26}

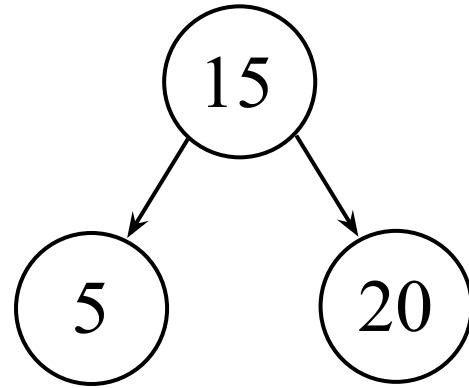
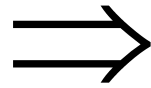
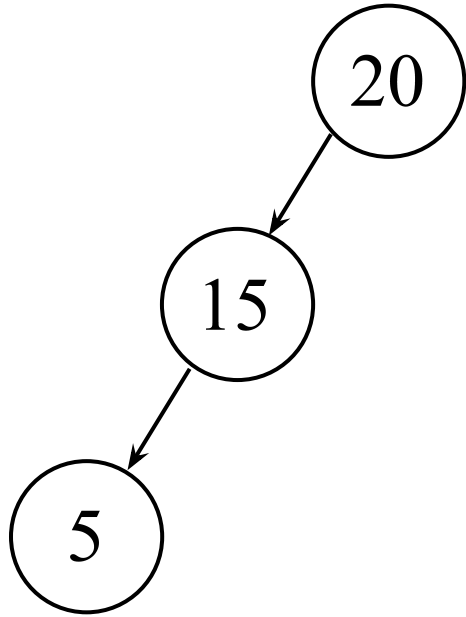
Шаг 1:



Шаг 2:

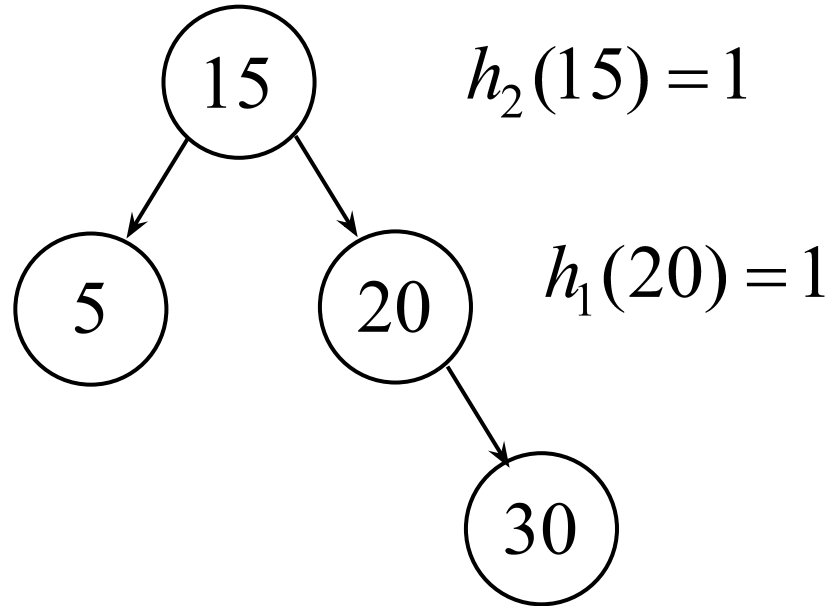
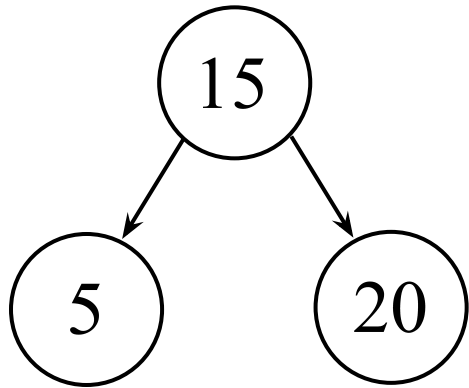


$$h_1 h_2 > 0, \quad h_1 < 0 \quad \Rightarrow \quad R$$



{30, 55, 25, 10, 6, 2, 17, 35, 40, 27, 11, 26}

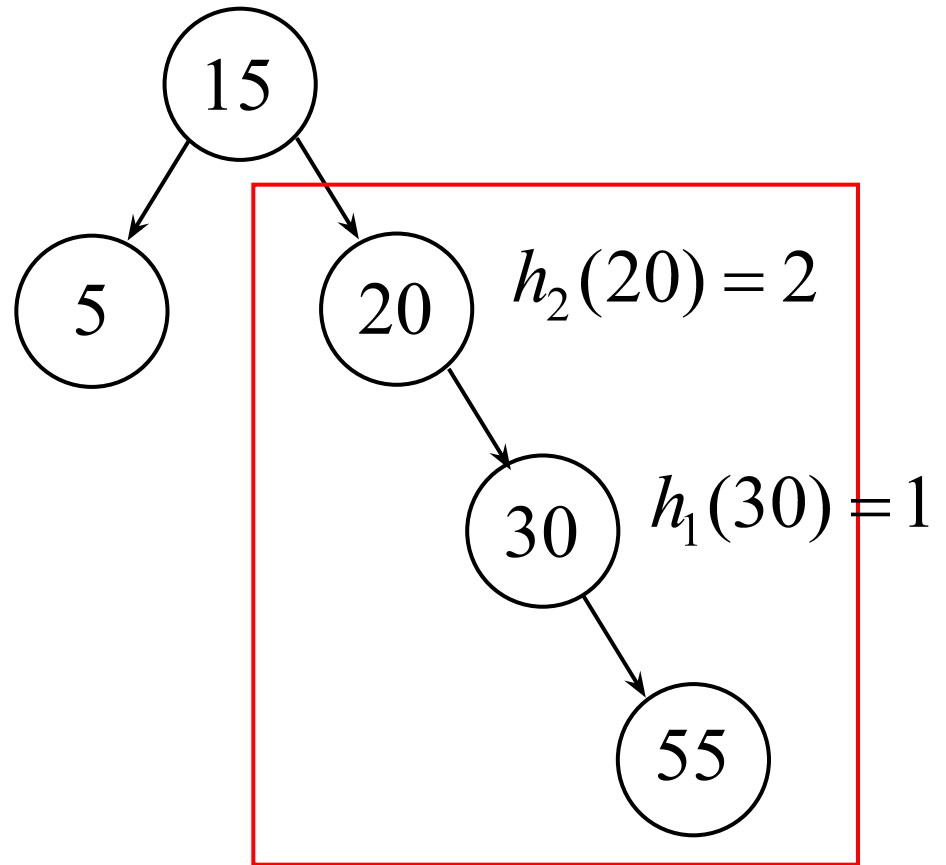
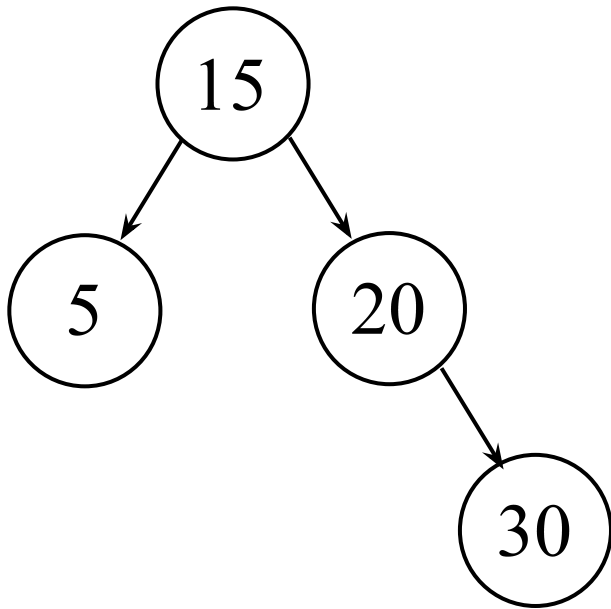
Шаг 3:



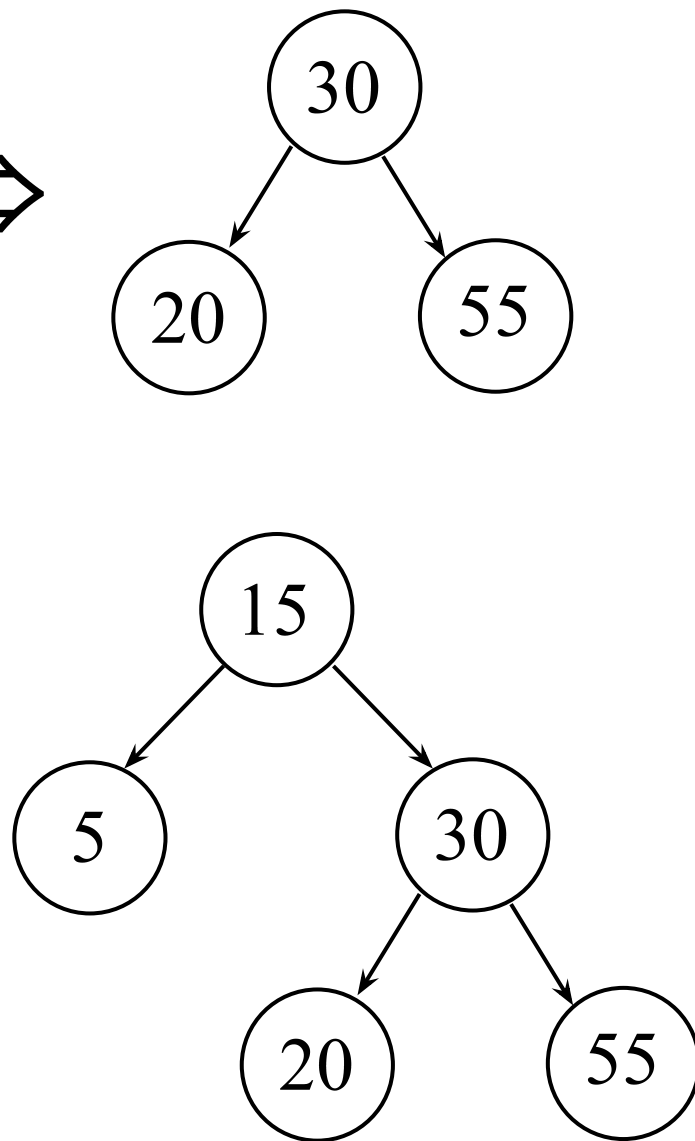
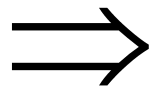
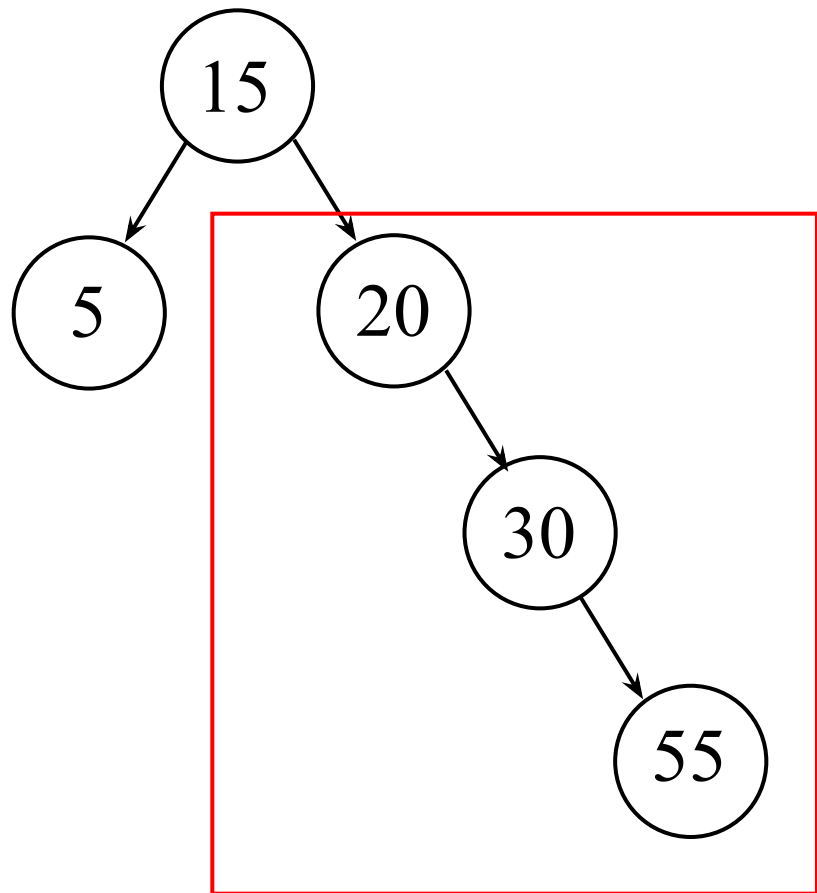
$\Rightarrow$  *балансировка не нужна*

{55, 25, 10, 6, 2, 17, 35, 40, 27, 11, 26}

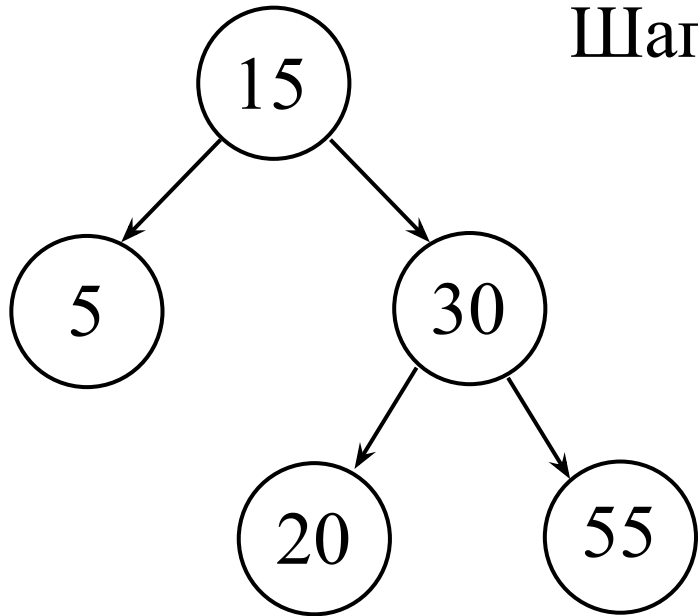
Шаг 4:



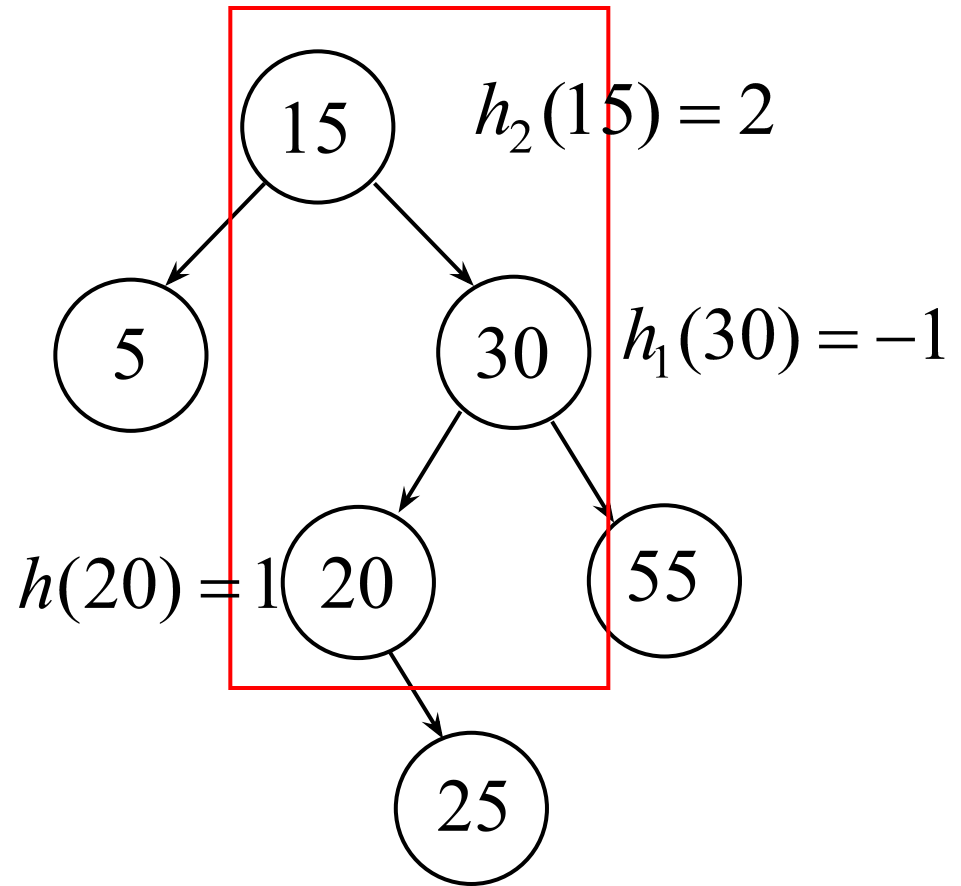
$$h_1 h_2 > 0, \quad h_1 > 0 \quad \Rightarrow \quad L$$



{25, 10, 6, 2, 17, 35, 40, 27, 11, 26}

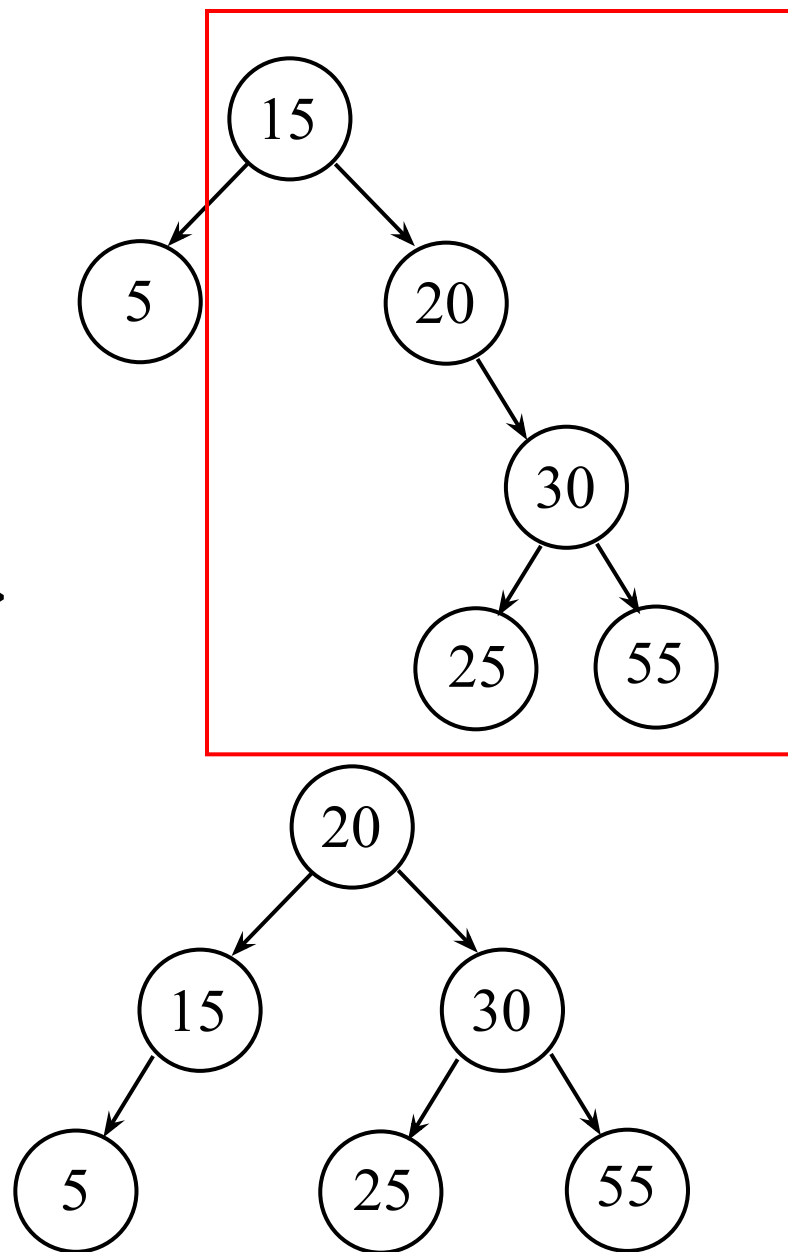
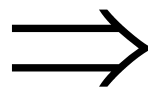
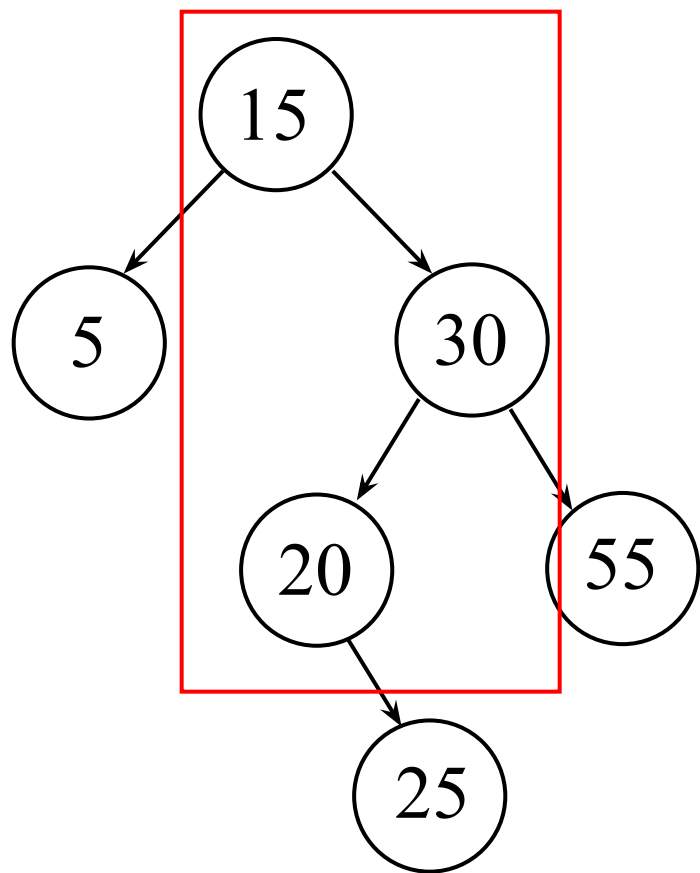


Шаг 5:



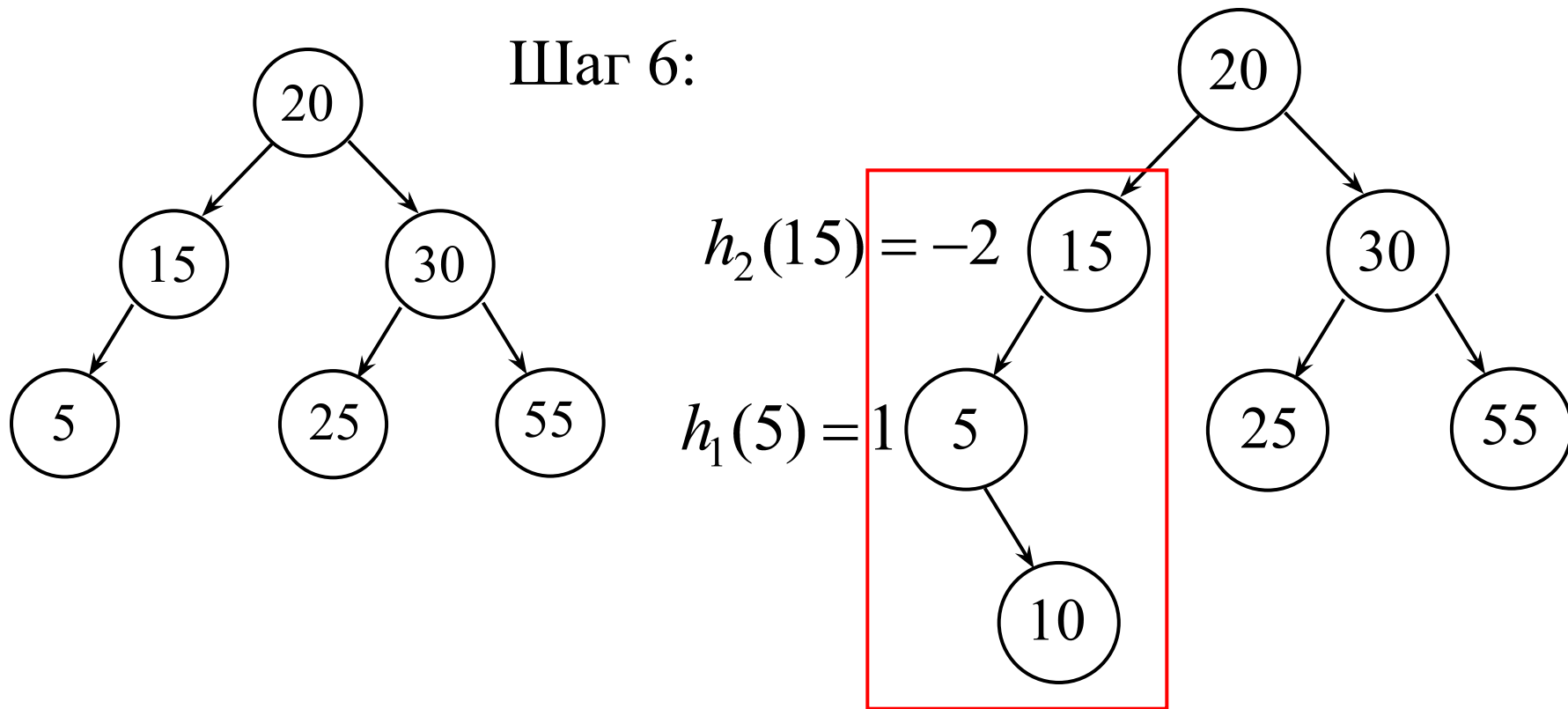
$$h_1 h_2 < 0, \quad h_1 < 0 \quad \Rightarrow \quad RL$$

Поворот RL:



{10, 6, 2, 17, 35, 40, 27, 11, 26}

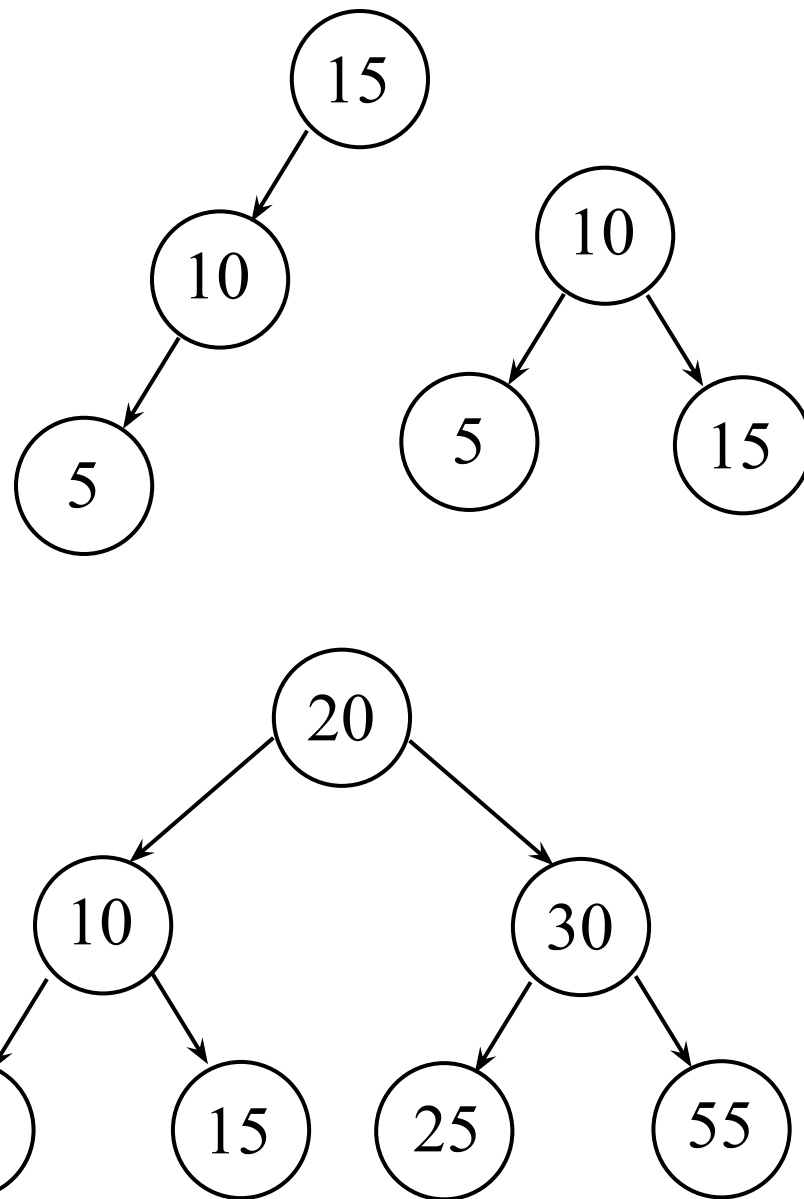
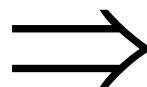
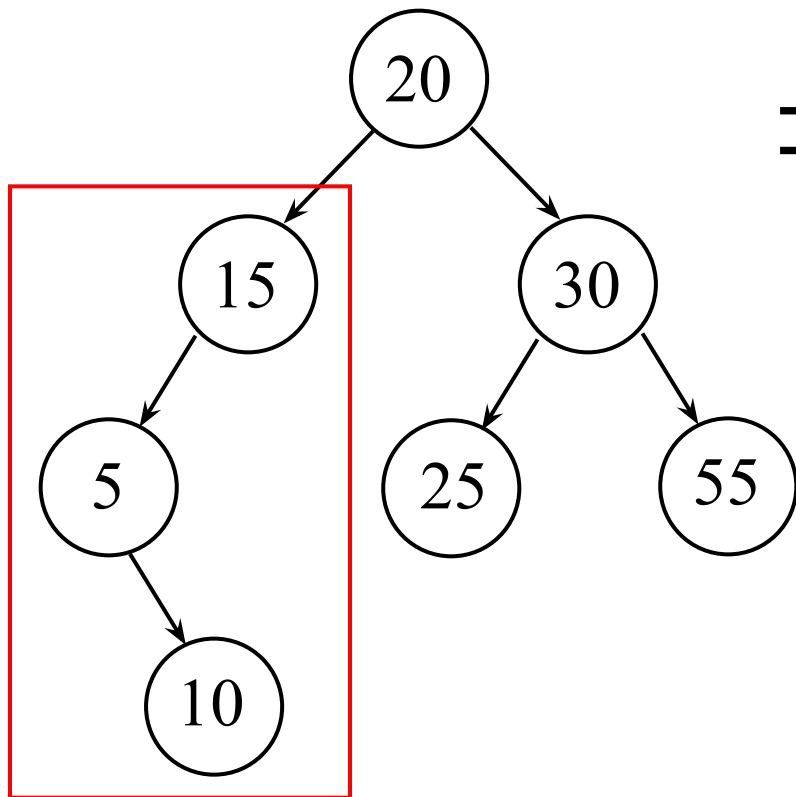
Шаг 6:



$$h_1 h_2 < 0, \quad h_1 > 0 \quad \Rightarrow \quad LR$$

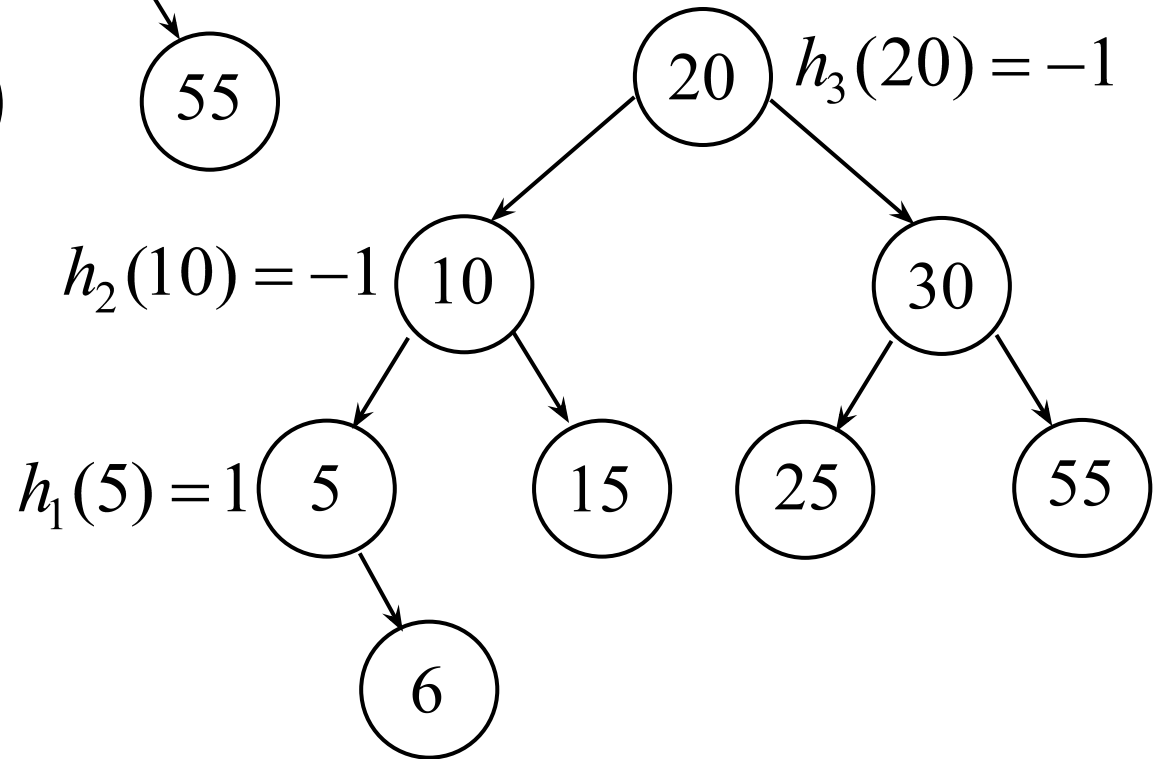
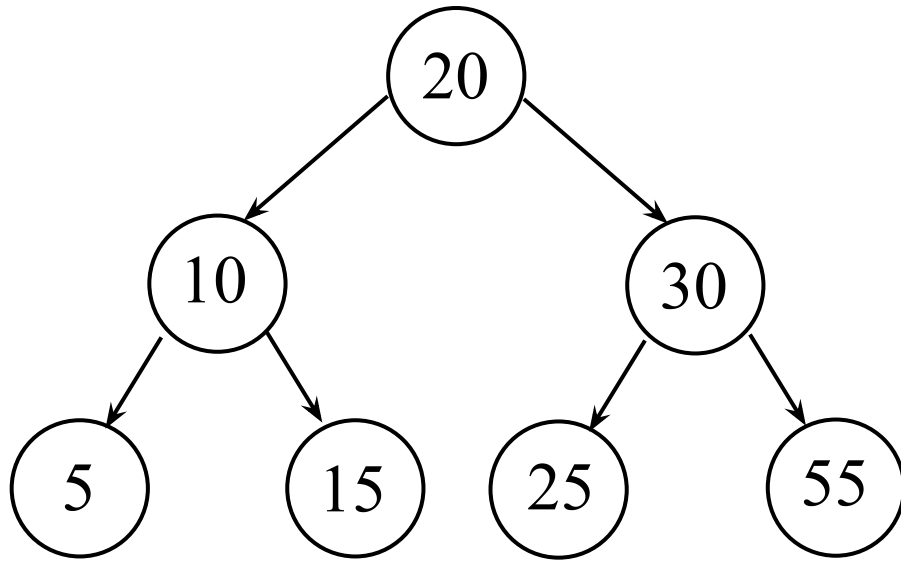


Поворот LR:



{6, 2, 17, 35, 40, 27, 11, 26}

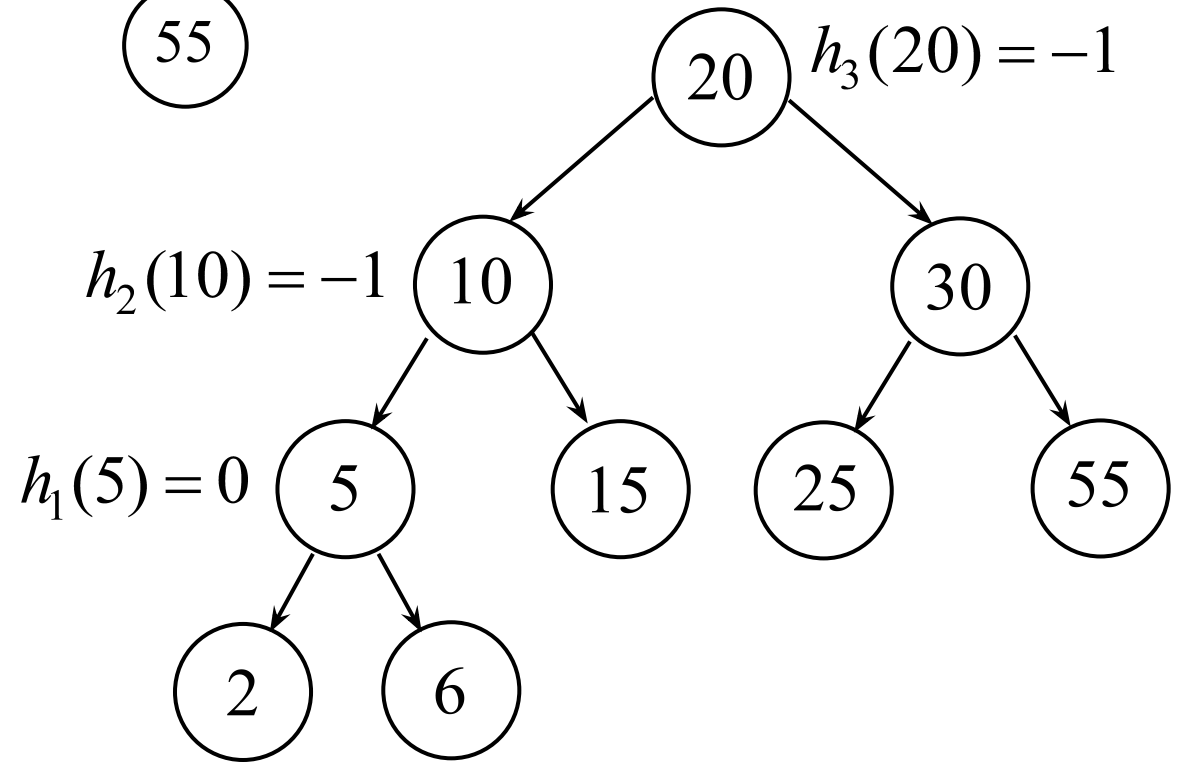
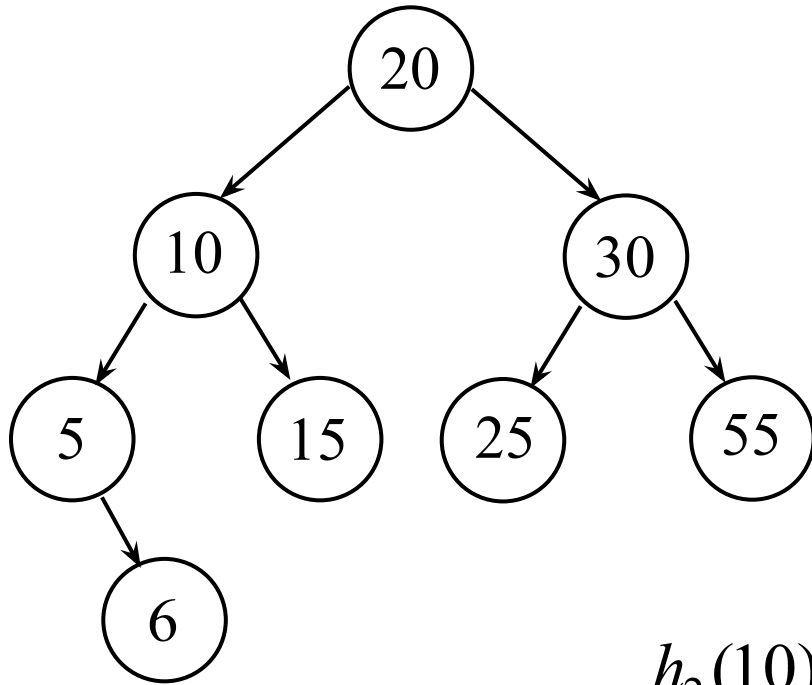
Шаг 7:



$\Rightarrow$  *балансировка не нужна*

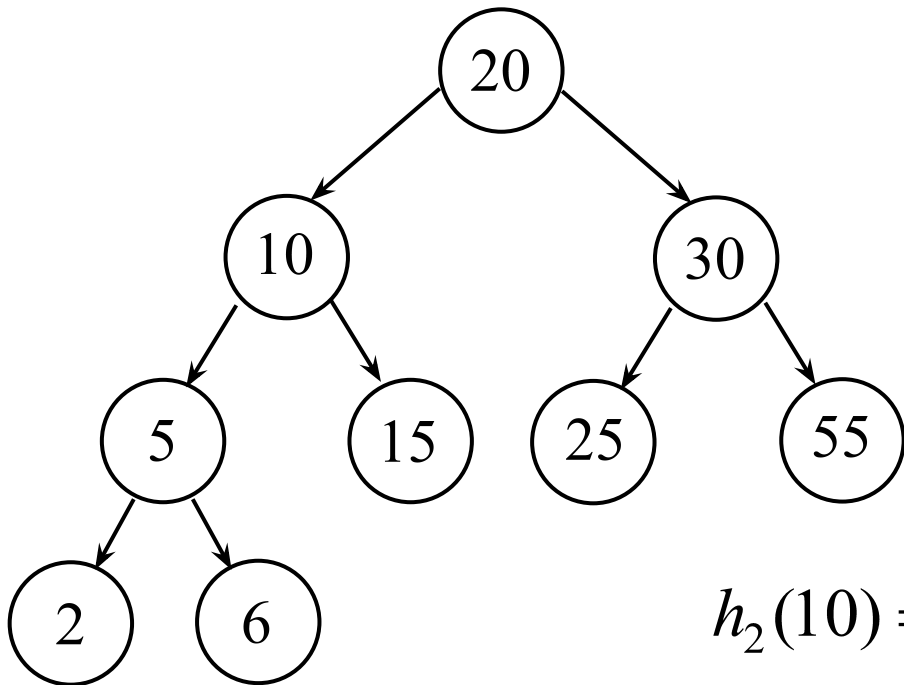
{2, 17, 35, 40, 27, 11, 26}

Шаг 8:

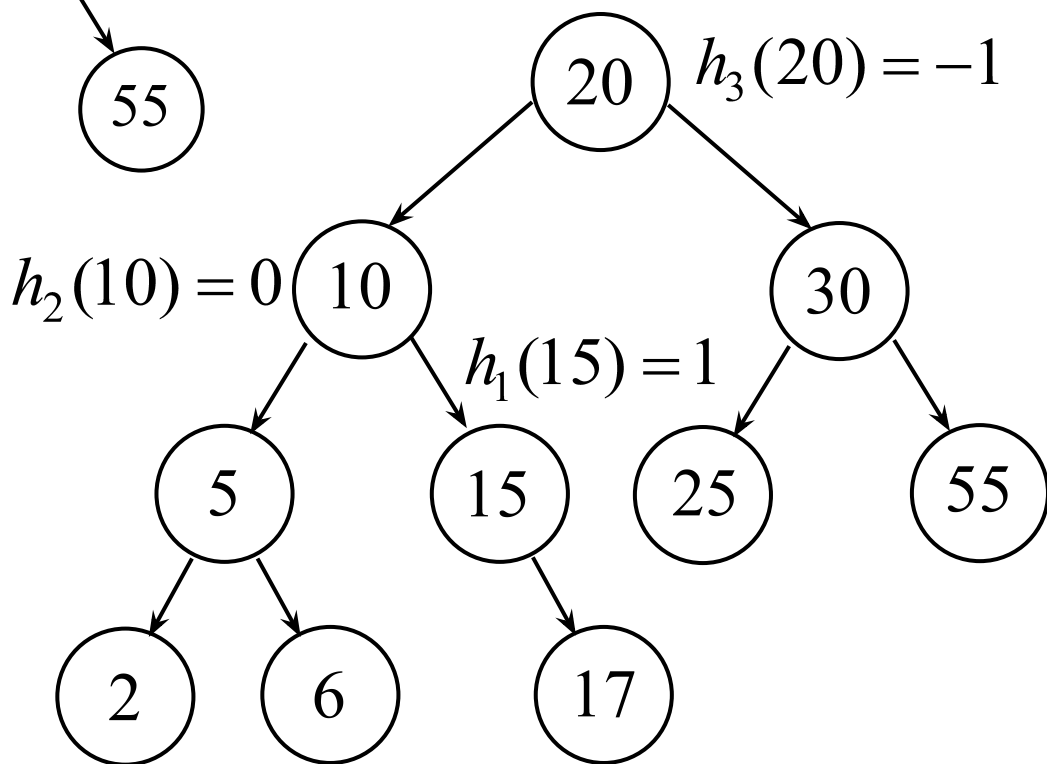


$\Rightarrow$  балансировка не нужна

{17, 35, 40, 27, 11, 26}



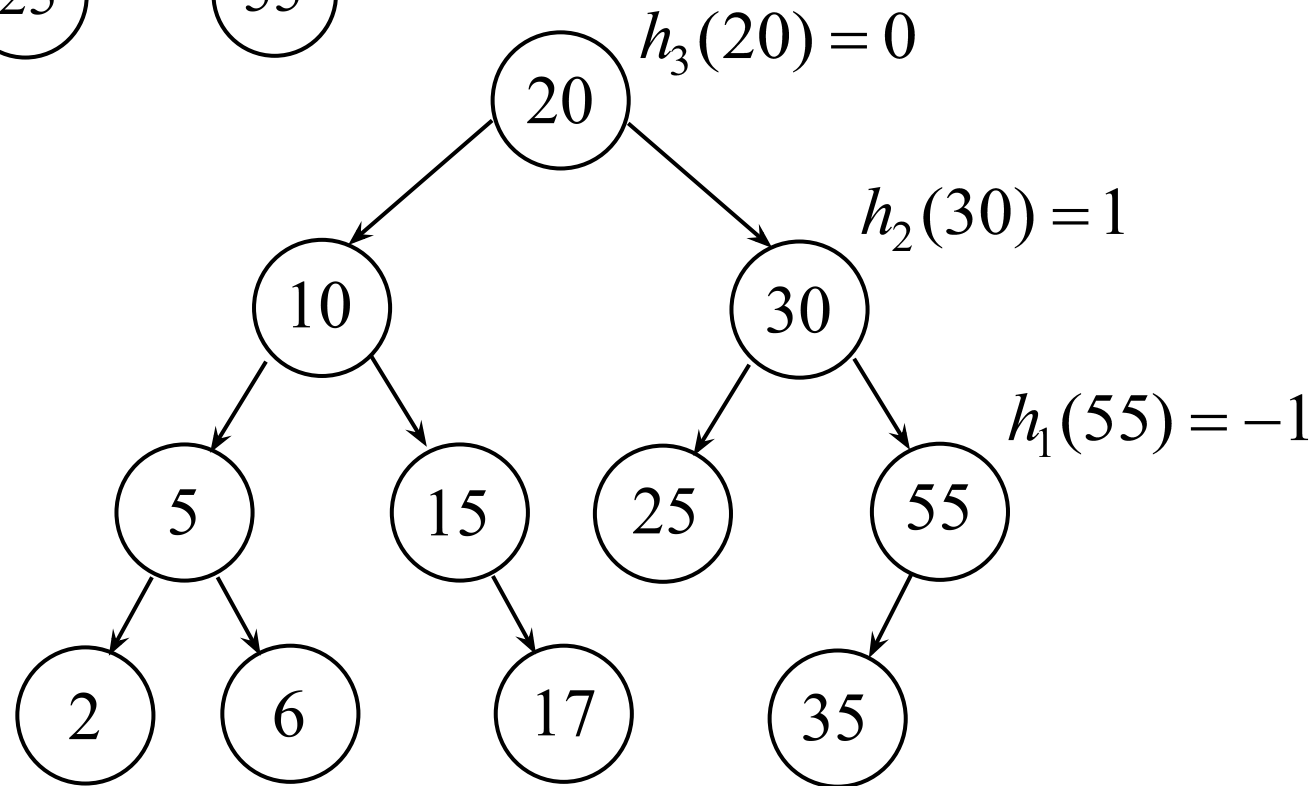
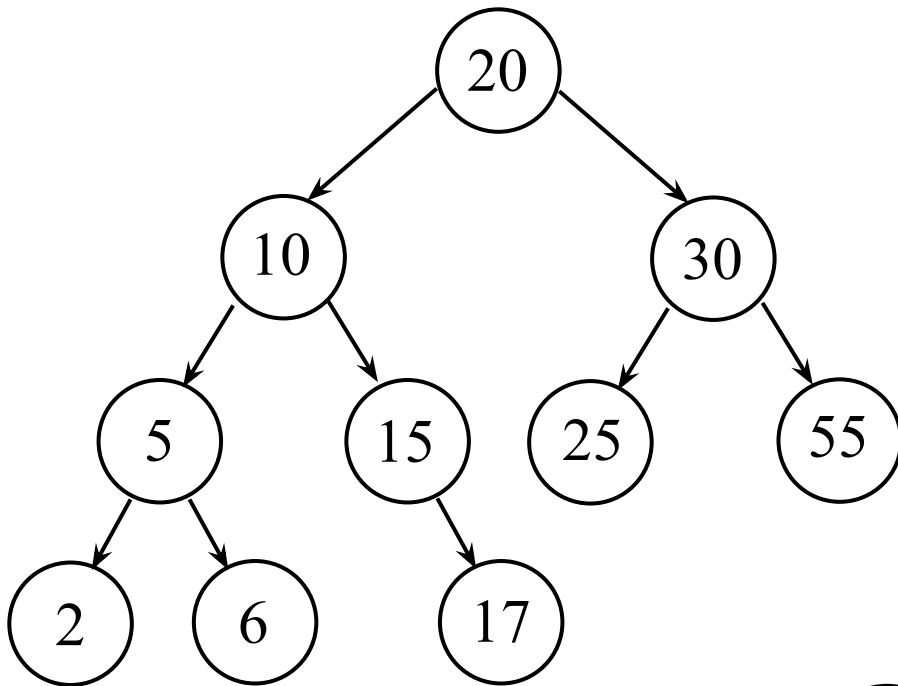
Шаг 9:



⇒ *балансировка не нужна*

{35, 40, 27, 11, 26}

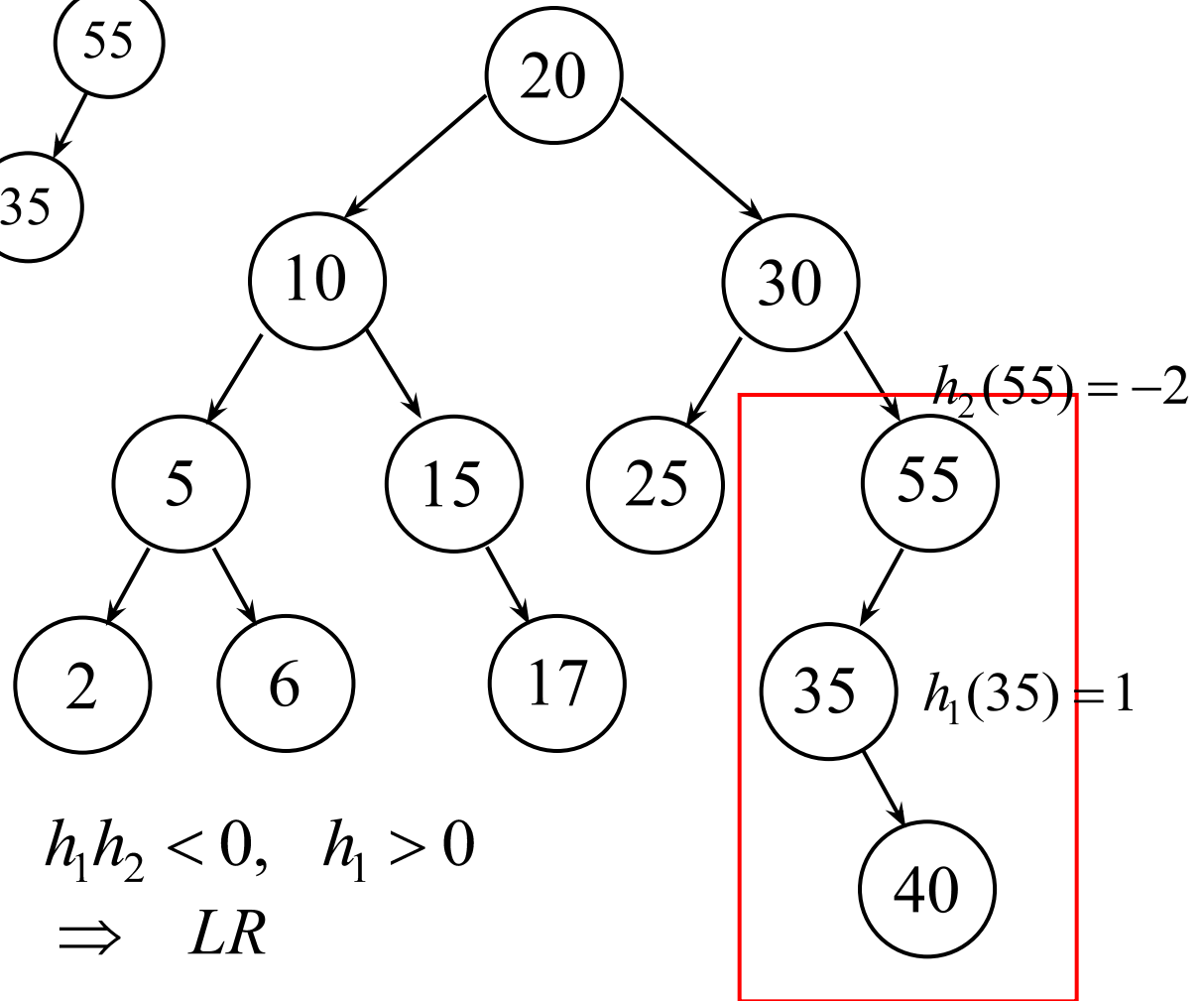
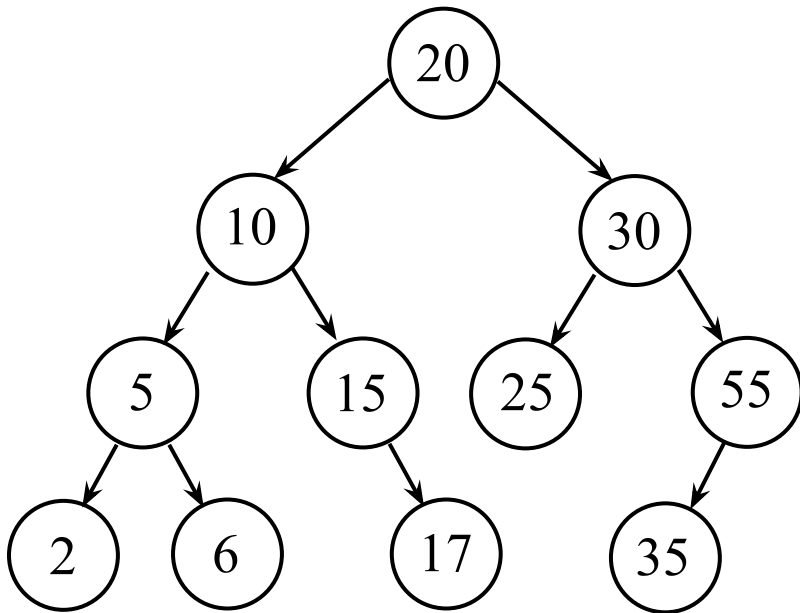
Шаг 10:



$\Rightarrow$  балансировка не нужна

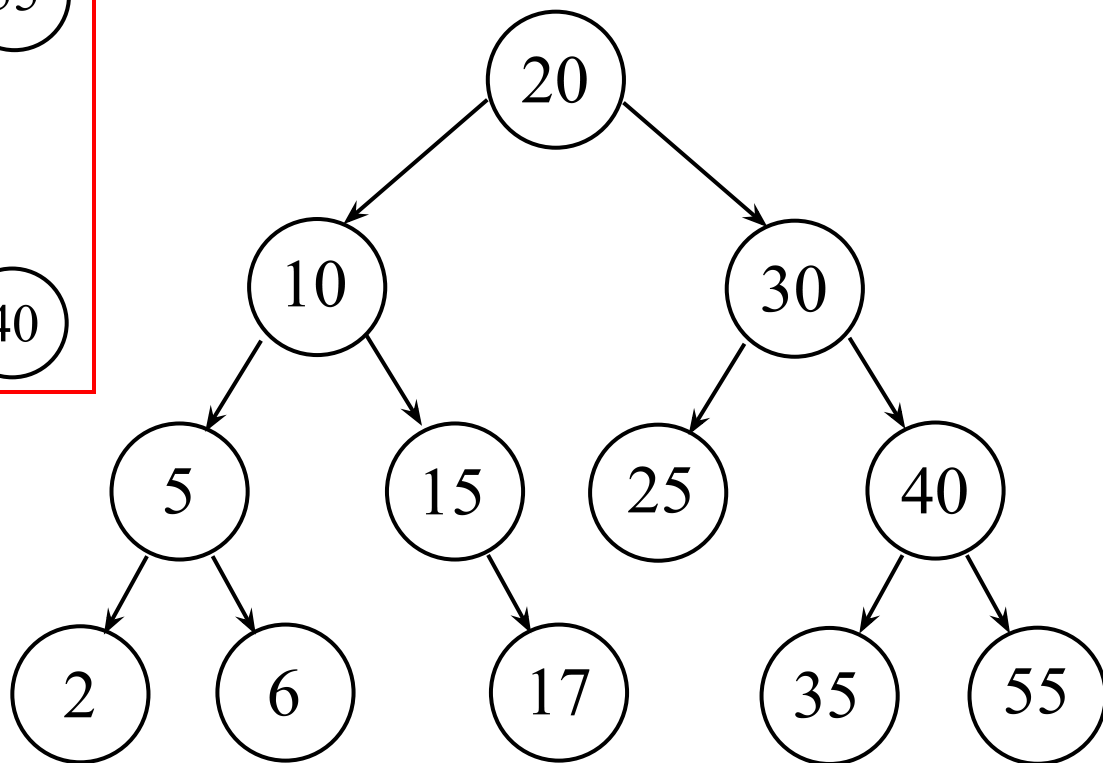
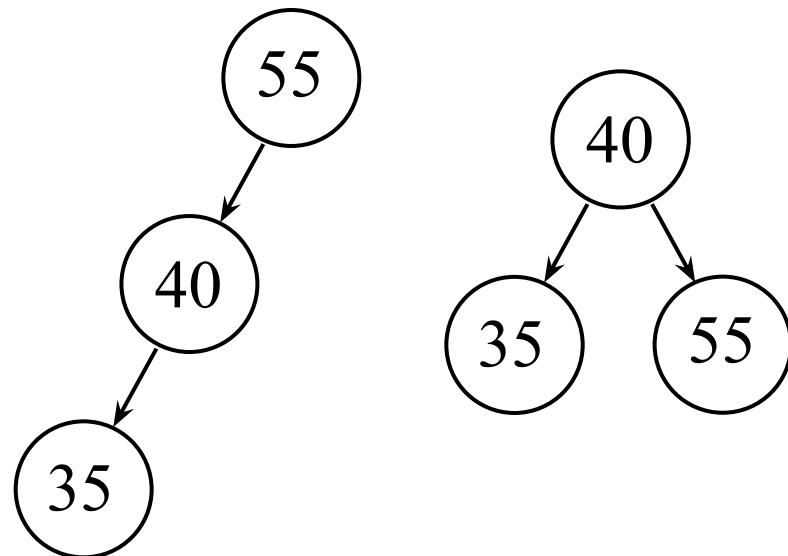
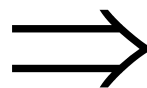
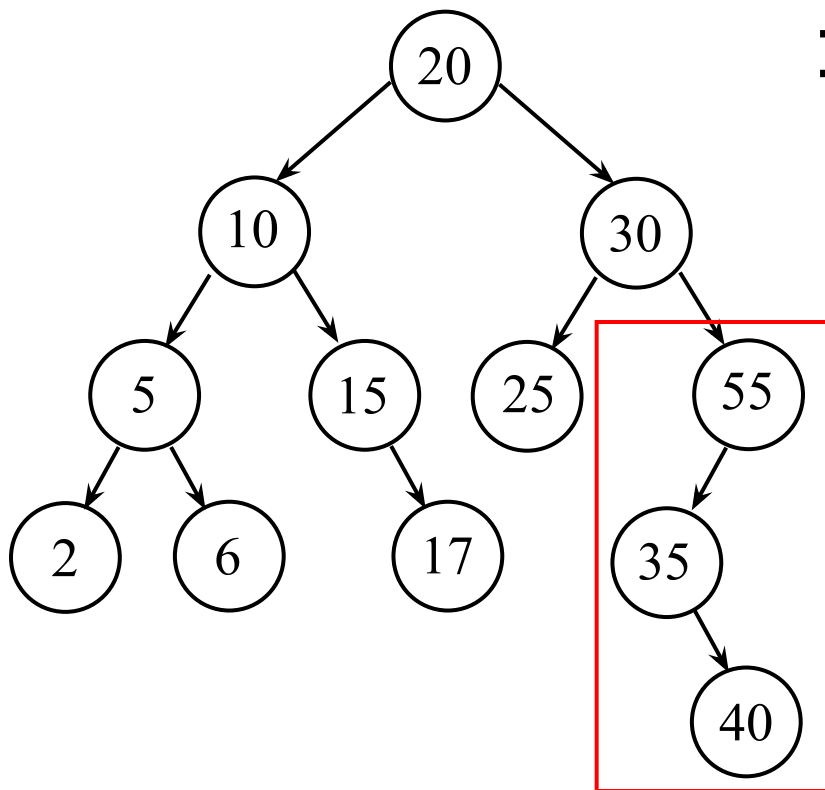
{ 40, 27, 11, 26 }

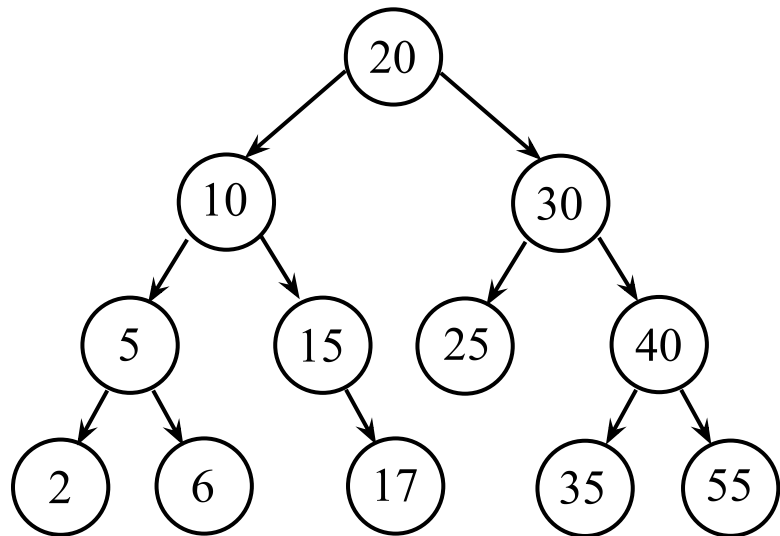
Шаг 11:



$$h_1 h_2 < 0, \quad h_1 > 0 \\ \Rightarrow LR$$

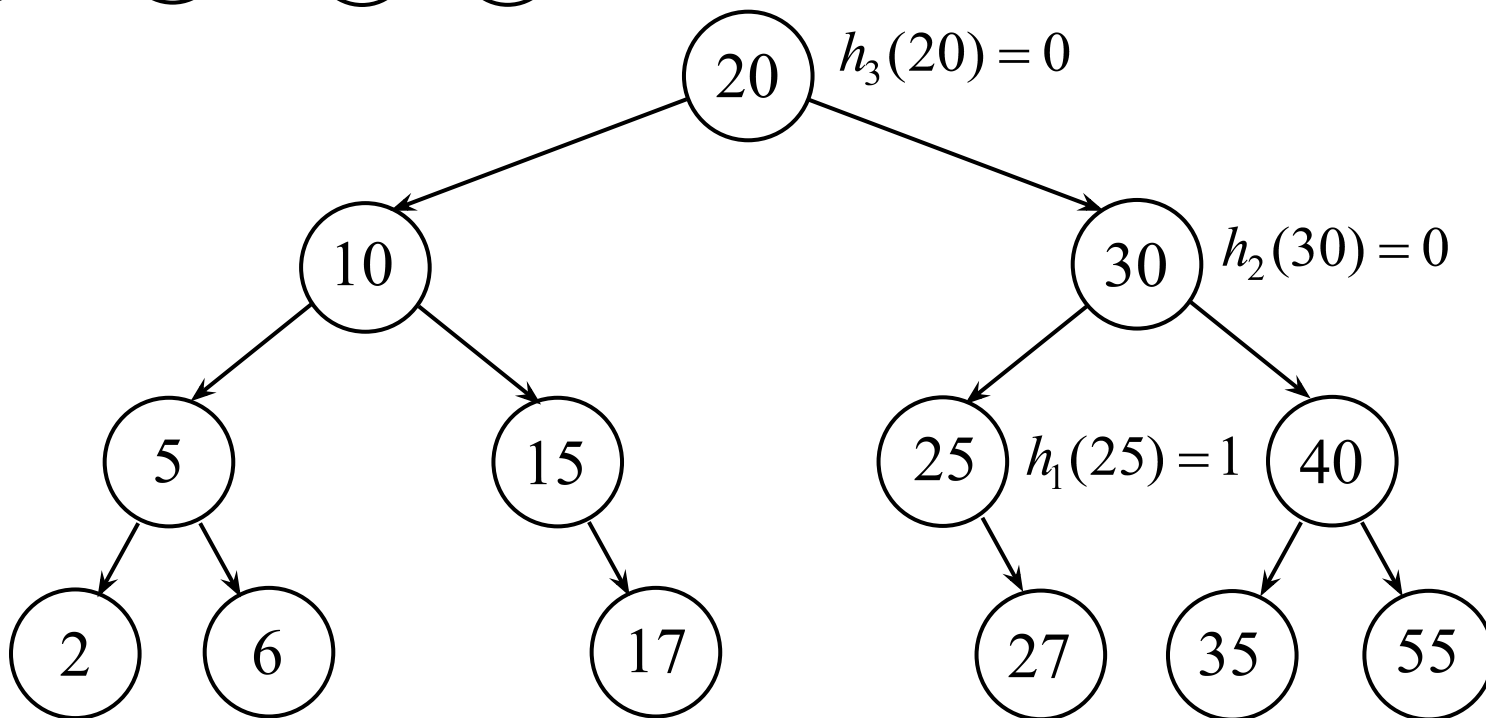
Поворот LR:





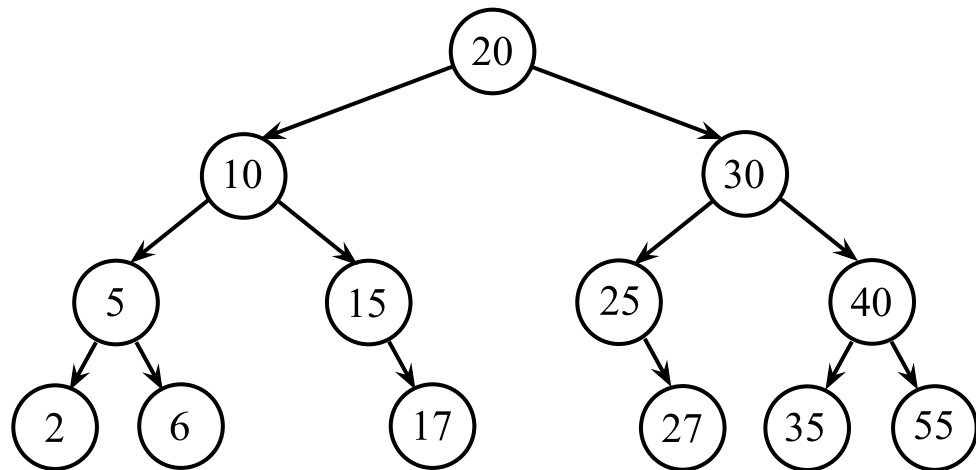
{27, 11, 26}

Шаг 12:



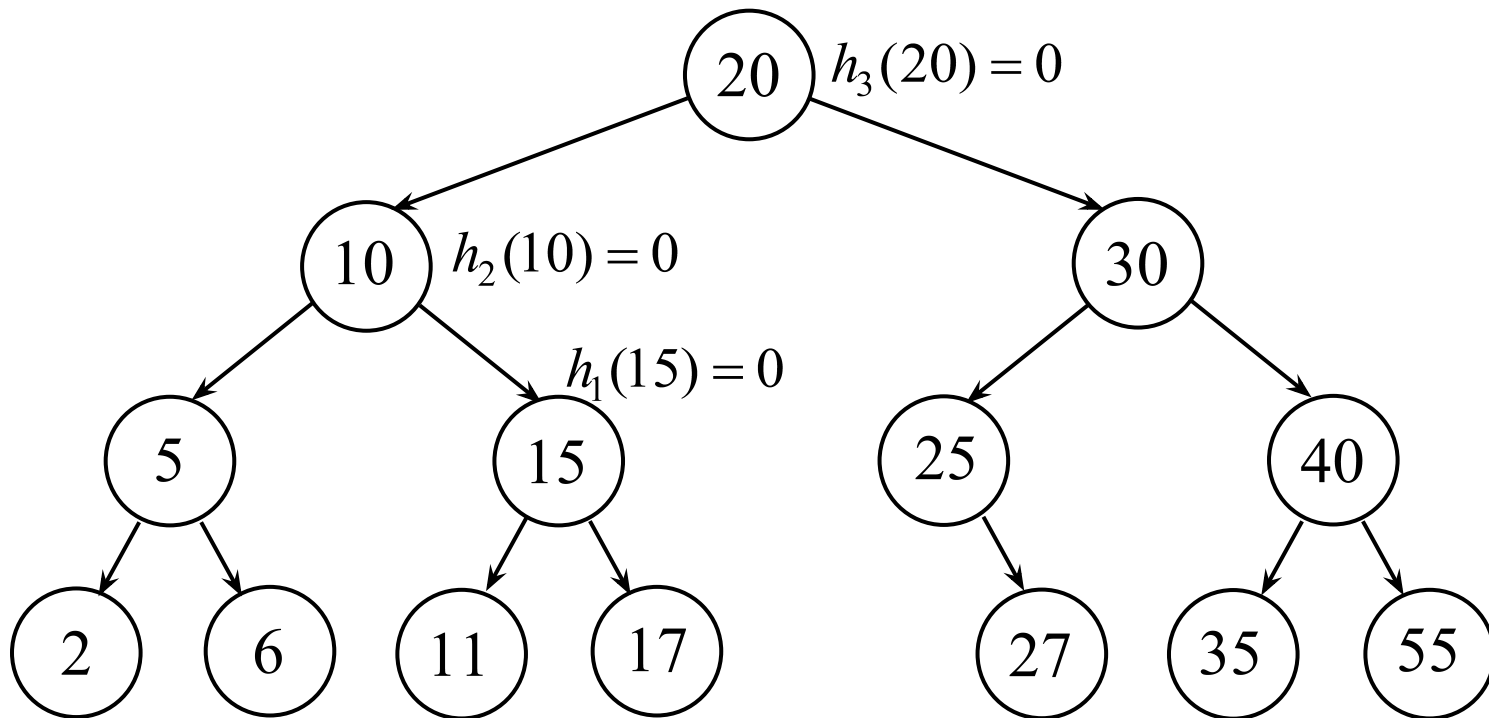
⇒ *балансировка не нужна*



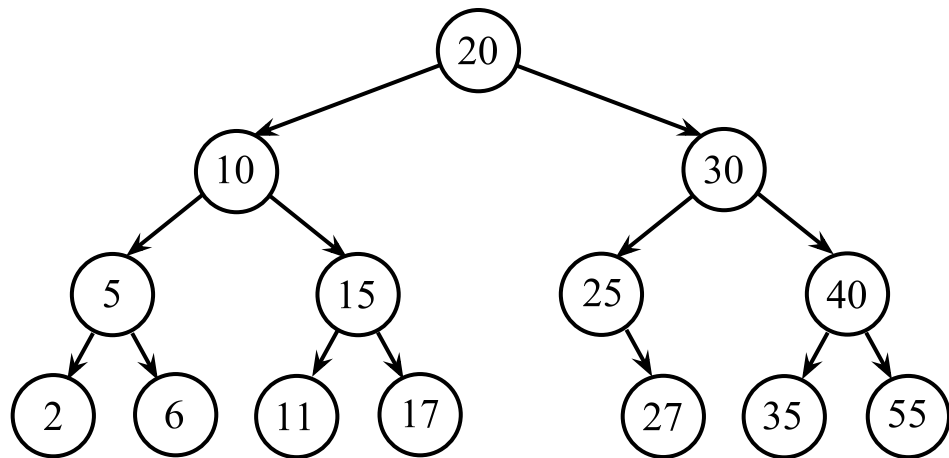


{11, 26}

Шаг 13:

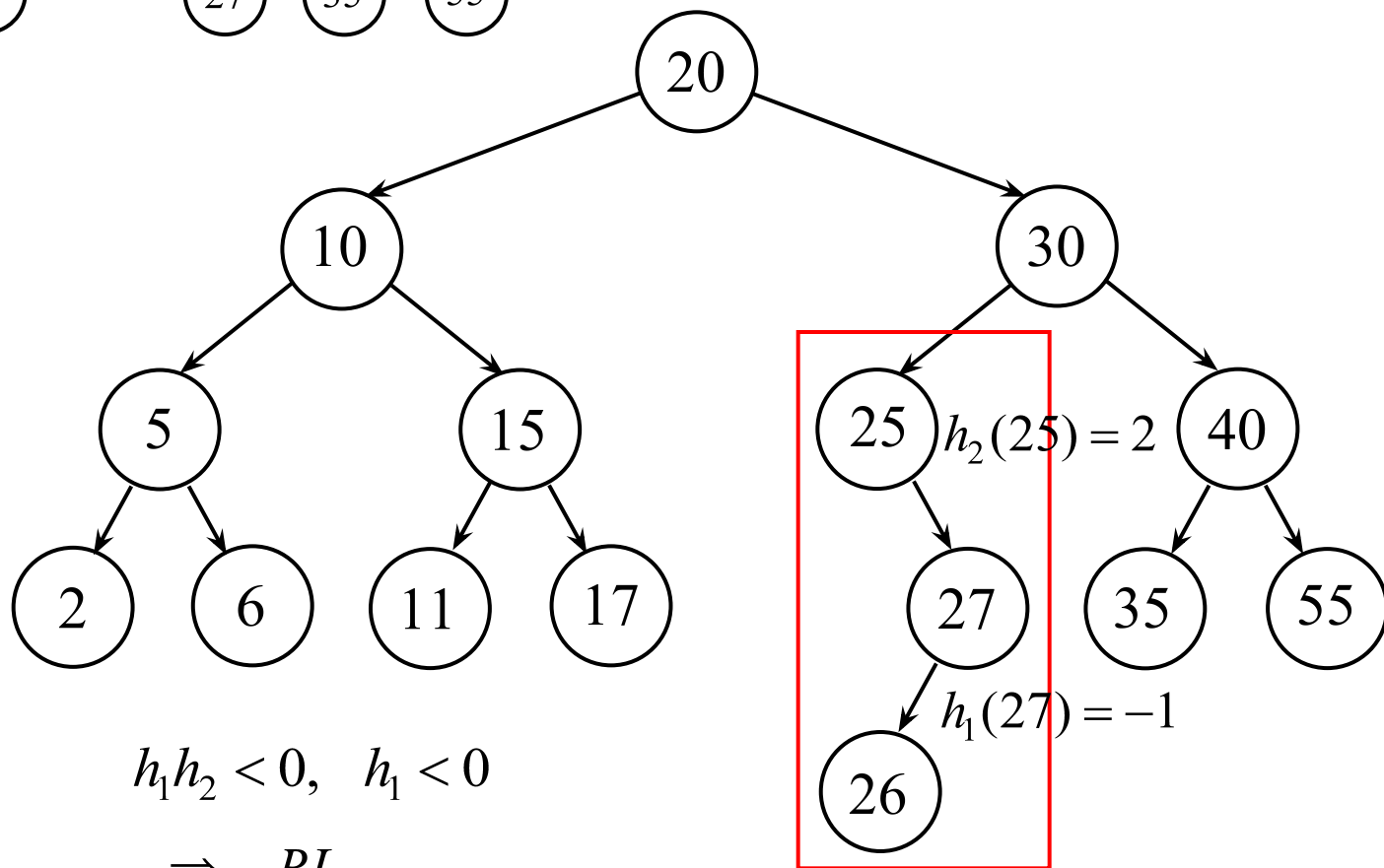


⇒ *балансировка не нужна*

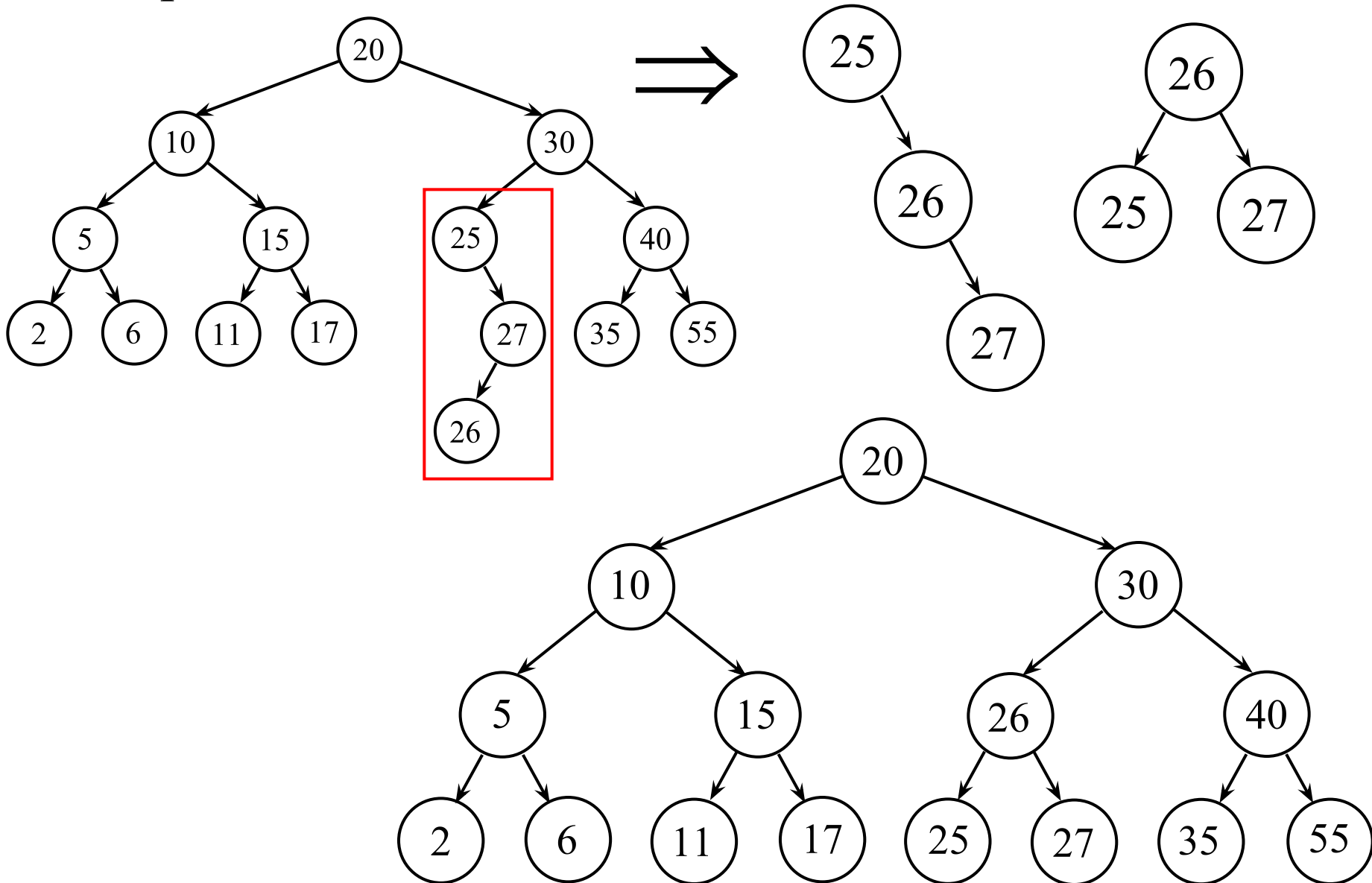


{26}

Шаг 14:



Поворот RL:



# Сбалансированное дерево поиска

