

- drzewo binarne
- po co?
- balansowanie po operacjach:  
wstawiania i usuwania



Georgij Adelson-Wielski

Доклады Академии наук СССР  
1962. Том 146, № 2

МАТЕМАТИКА

Г. М. АДЕЛЬСОН-ВЕЛЬСКИЙ, Е. М. ЛАНДИС

**ОДИН АЛГОРИТМ ОРГАНИЗАЦИИ ИНФОРМАЦИИ**

(Представлено академиком И. Г. Петровским 17 IV 1962)

В заметке будет идти речь об организации информации, расположенной в ячейках автоматической вычислительной машины. Для определенности будет рассматриваться трехадресная машина.

Постановка задачи. В машину последовательно поступает информация из некоторого запаса. Элемент информации содержится в группе ячеек, расположенных подряд. В элементе информации содержится некоторое число — оценка информации, — различное для различных элементов. Требуется организовать размещение информации в памяти машины так, чтобы в любой момент поиск информации с данной оценкой и занесение нового элемента информации требовали не очень большого числа действий.

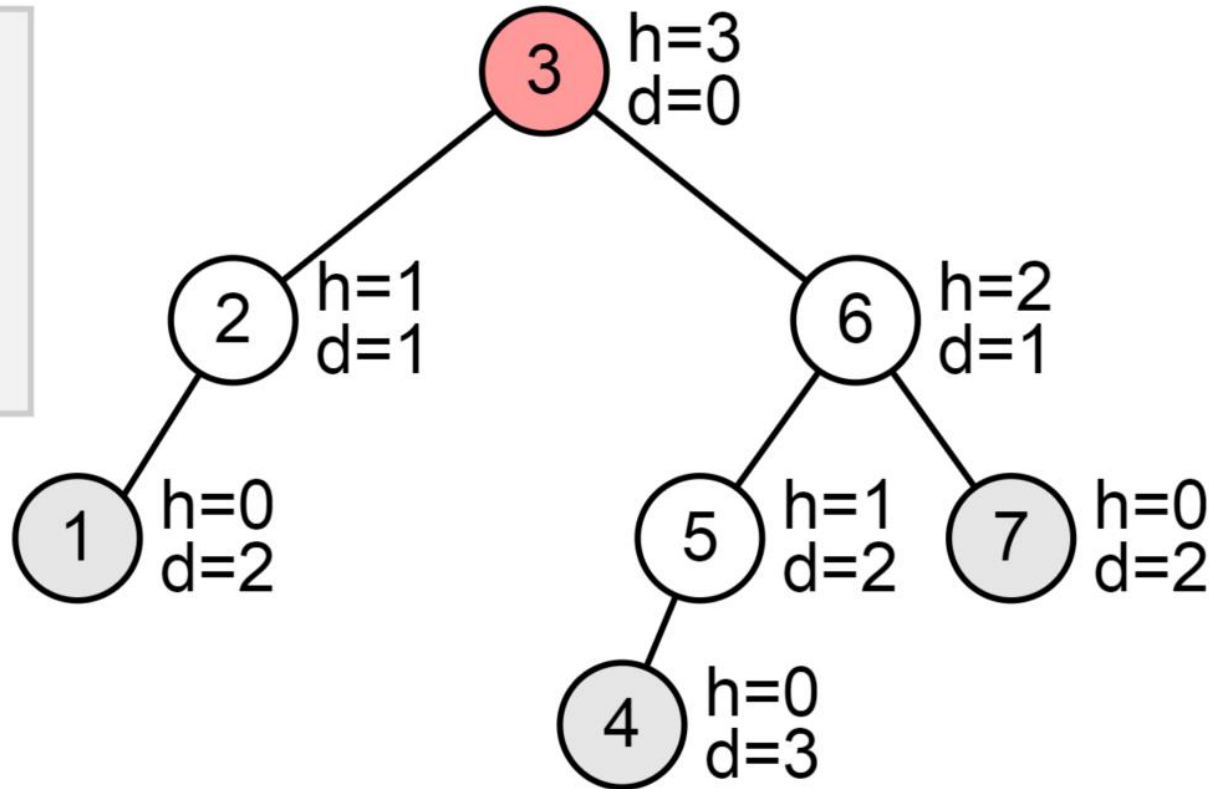
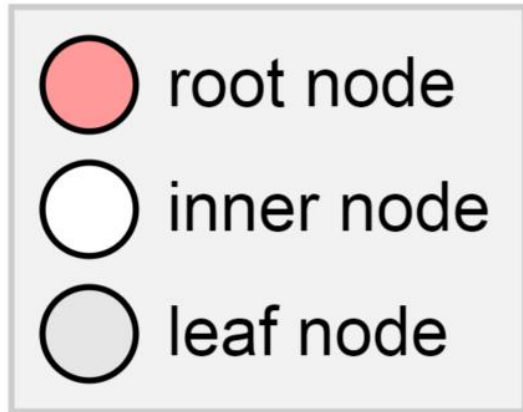
В заметке предлагается алгоритм, где как поиск, так и занесение производятся за  $C \lg N$  действий, где  $N$  — число элементов информации, поступивших к данному моменту.

Для хранения поступающей информации отводится часть памяти машины.



Jewgienij Łandis

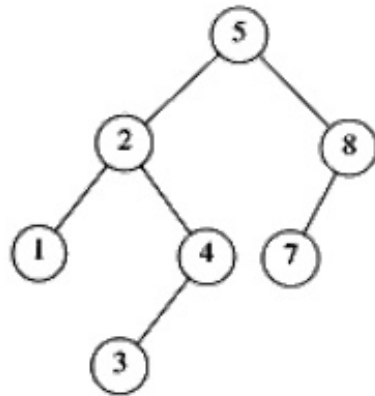
# Głębokość i wysokość



- Balansowanie: różnica wysokości lewego i prawego dziecka musi być  $\leq 1$ .

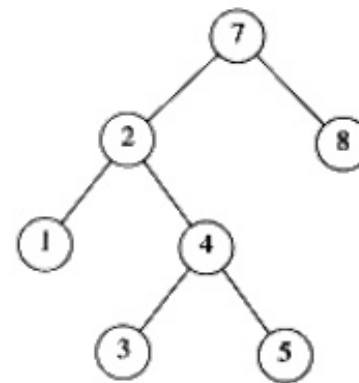
# Czym jest więc drzewo AVL?

## AVL tree?



**YES**

*Each left sub-tree has height 1 greater than each right sub-tree*



**NO**

*Left sub-tree has height 3, but right sub-tree has height 1*

# Przykładowe operacje

- Wyszukiwanie
- Przechodzenie
- Wstawianie
- Usuwanie
- Rotacja pojedyncza w prawo - **LL** (ang. Right-Right)
- Rotacja pojedyncza w lewo - **RR** (ang. Left-Left)
- Rotacja podwójna w prawo-lewo - **RL** (ang. Right-Left)
- Rotacja podwójna w lewo-prawo - **LR** (ang. Left-Right)

# Jak to działa?

