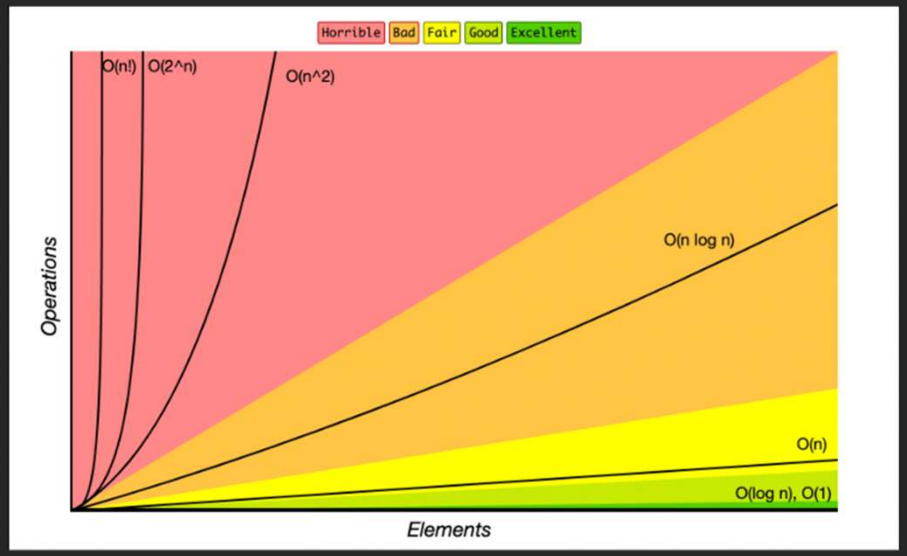

PROGRAMMING ADVICES
LEARN THE RIGHT WAY
26+ Years of Experience
Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CSE, ITILF, MCPD, MCSD
لا تنسى الاشتراك في قناتنا على اليوتيوب ومشاركة القناة مع اصدقائك
لتعم الفائدة للجميع وانقاذ الاف الناس من التشتت وجزاكم الله خيرا
لا تنسونا من دعائكم وادعو لوالدي بالرحمة
www.ProgrammingAdvices.com


مهم جداً
هذا الملف للمراجعة السريعة واخذ الملاحظات عليه فقط ،لانه يحتوي
على اقل من 20٪ مما يتم شرحه في الفيديوهات
الاستعجال والاعتماد عليه فقط سوف يجعلك تخسر كميه معلومات
وخبرات كثيره
يجب عليك مشاهدة فيديو الدرس كاملا
لاتنسى عمل لايك ومشاركة القناة لتعم الفائدة للجميع
لا تنسونا من دعائكم
ProgrammingAdvices.com
Mohammed Abu-Hadhoud
STOP
PROGRAMMING
ADVICES
LEARN THE
RIGHT WAY


Algorithms & Problem Solving
Level 6
Introduction to Rotations
Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD
ProgrammingAdvices.com
PROGRAMMING ADVICES
LEARN THE RIGHT WAY

BST Search.

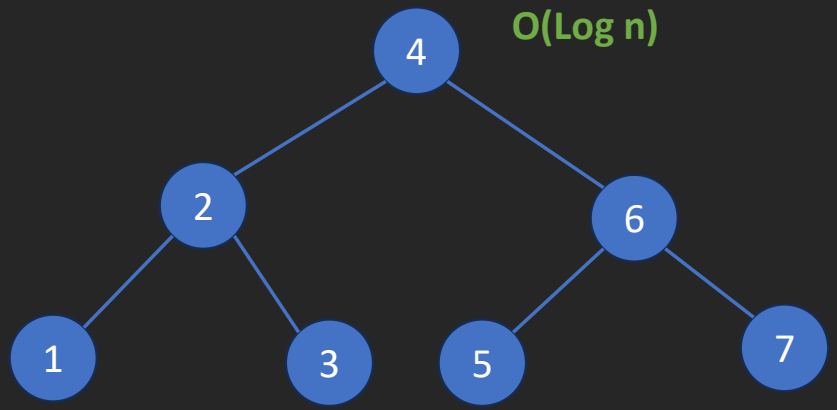
- Best/Average Case is O(Log n)
- Worst Case O(n)



Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

Balanced vs Imbalanced BST

Suppose that we entered data in BST in the following order:
4,2,6,1,3,5,7



Balanced

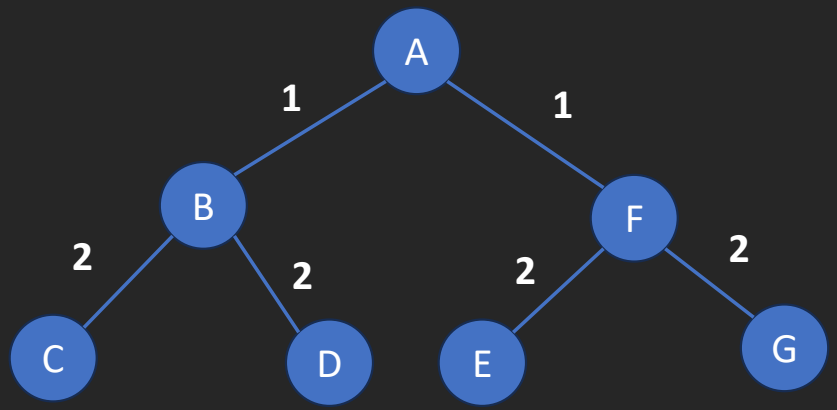
Suppose that we entered data in BST in the following order:
1,2,3,4,5,6,7

Unbalanced

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

Balanced Or Imbalanced?

BF = 2 – 2 = 0



Balanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

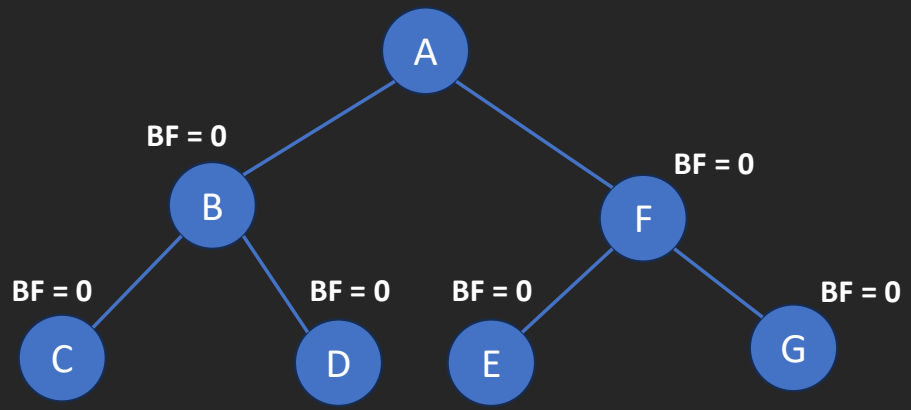
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

BF for each Node

BF = 0



Balanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

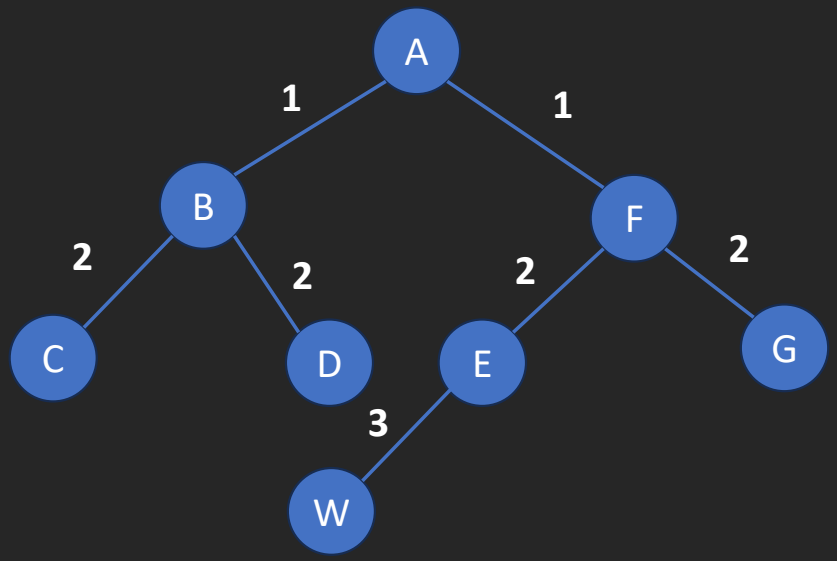
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

Balanced Or Imbalanced?

BF = 2 – 3 = -1



Balanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

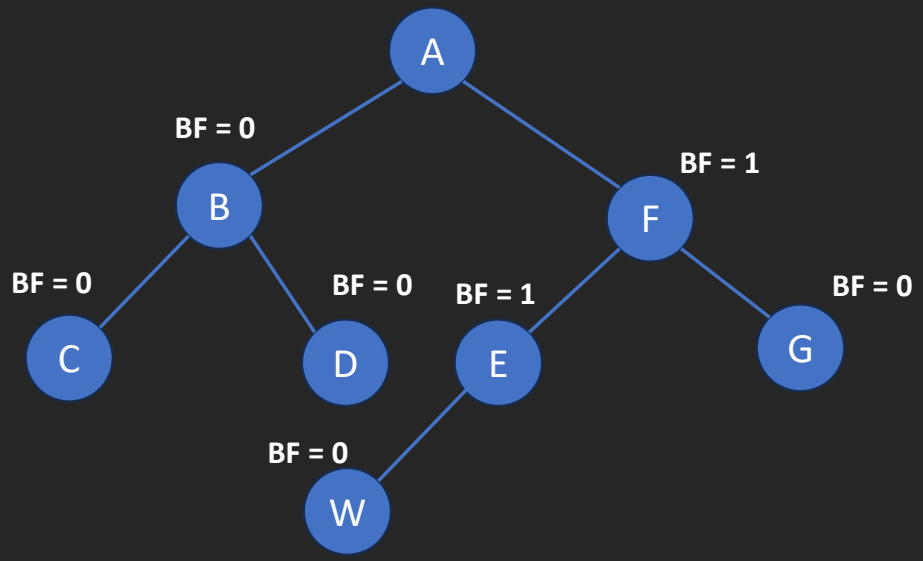
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

BF for each node

BF = 2 – 3 = -1



Balanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

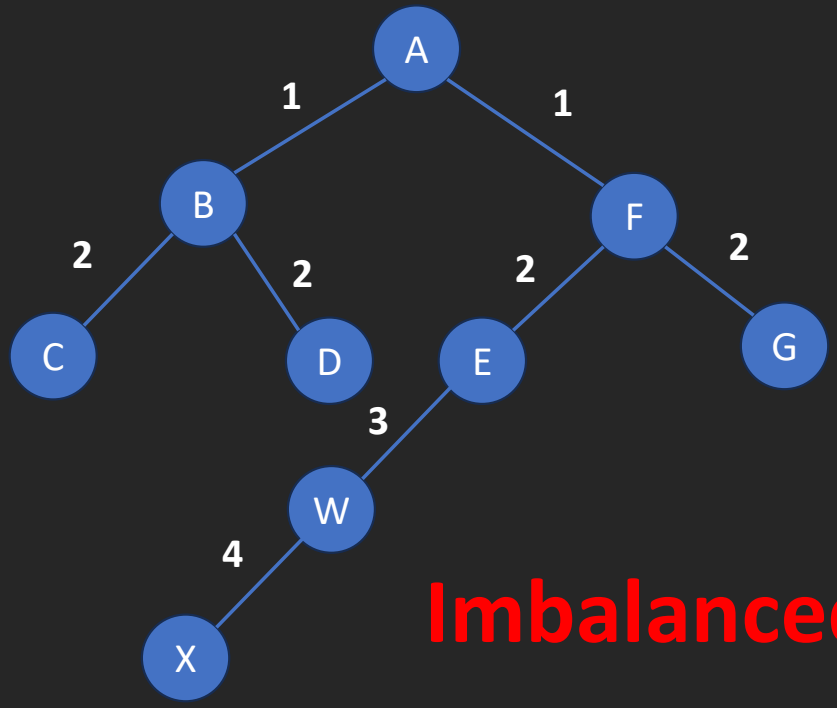
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

Balanced Or Imbalanced?

BF = 2 – 4 = -2



Imbalanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

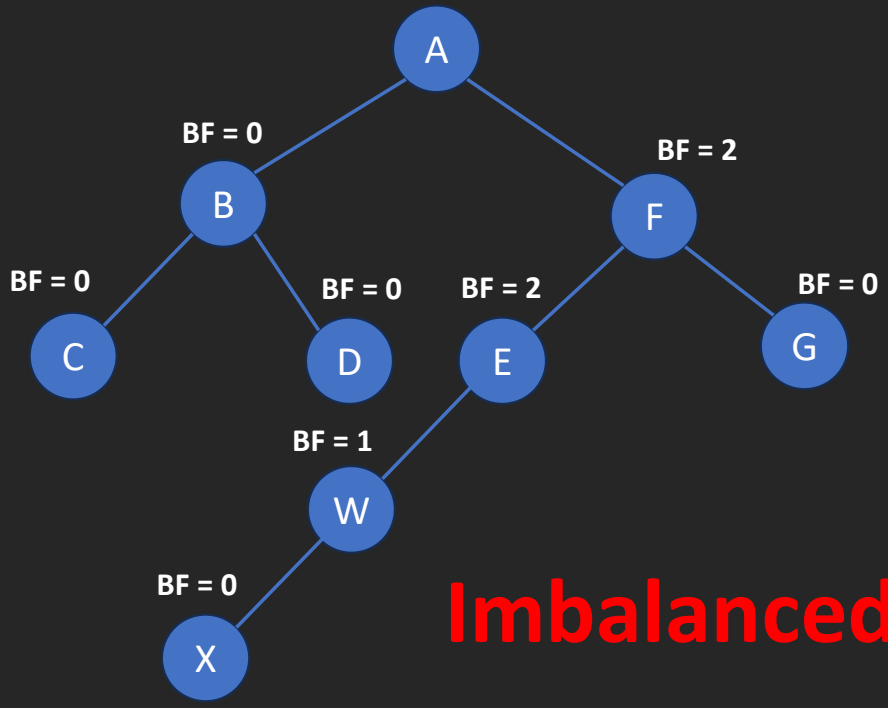
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

BF for each node

BF = 2 – 4 = -2



Imbalanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

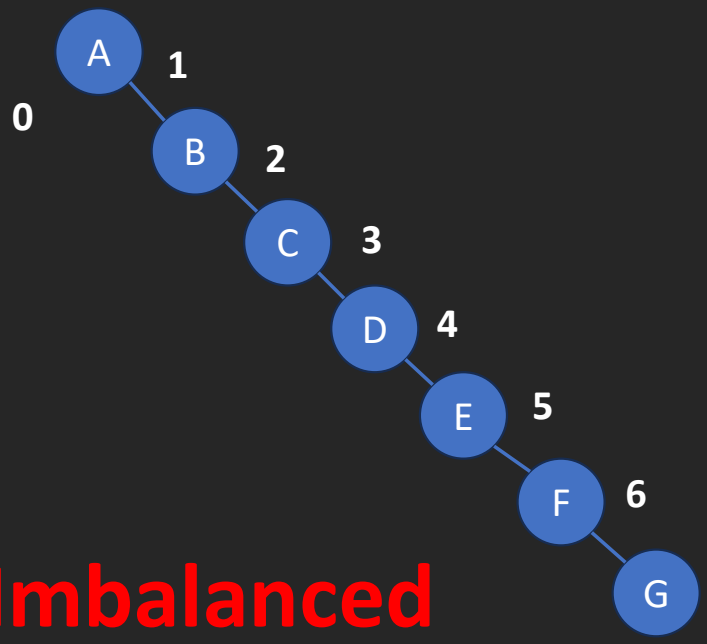
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

Balanced Or Imbalanced?

BF = 0 – 6 = -6



Imbalanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

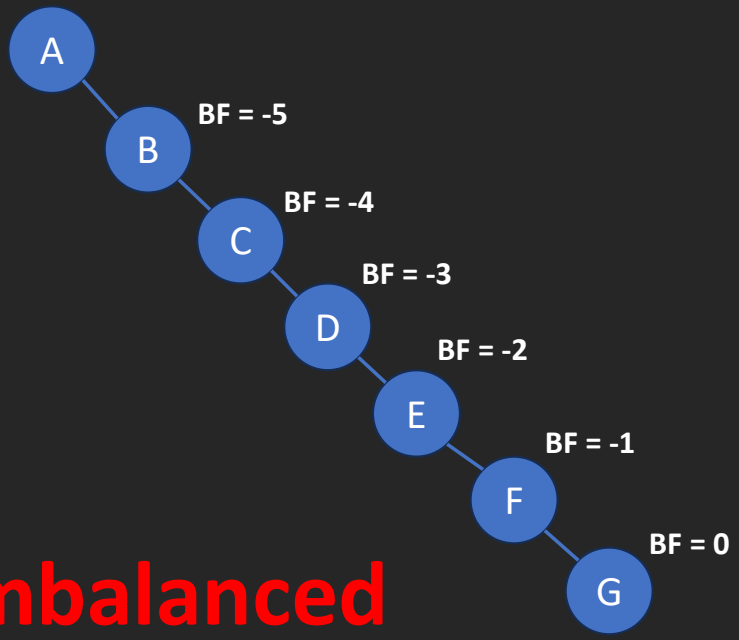
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

BF for each node

BF = 0 – 6 = -6



Imbalanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

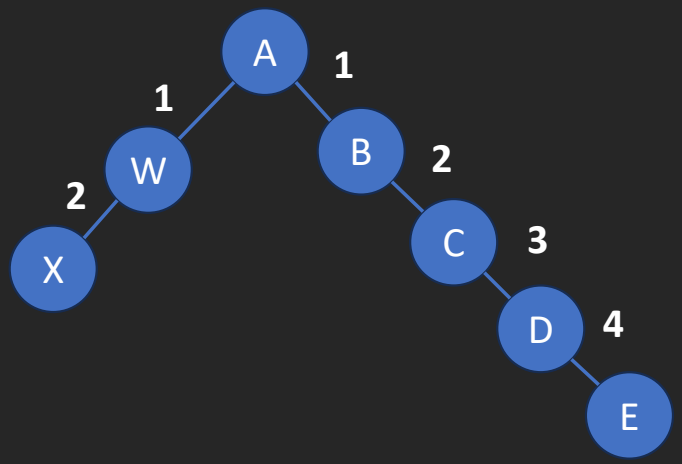
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

Balanced Or Imbalanced?

BF = 2 – 4 = -2



Imbalanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

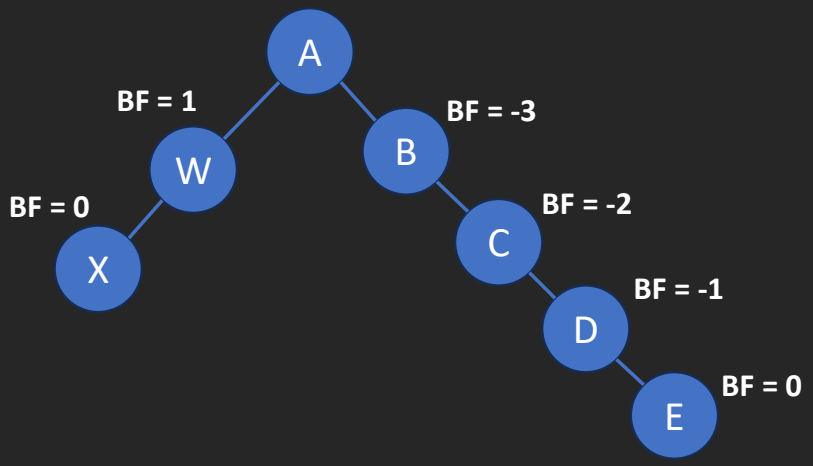
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

Balanced Or Imbalanced?

BF = 2 – 4 = -2



Imbalanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

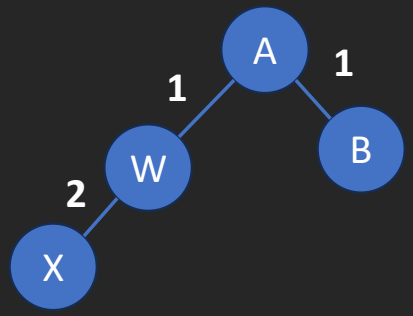
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

Balanced Or Imbalanced?

BF = 2 – 1 = 1



Balanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

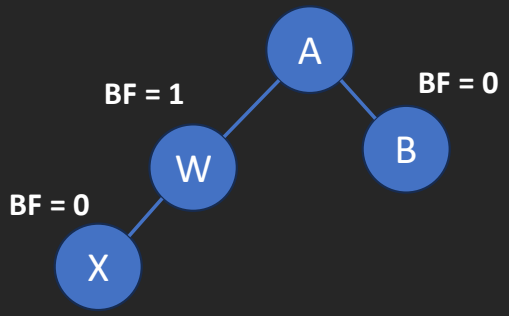
Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

BF for each node

BF = 2 – 1 = 1



Balanced

Rule:
Heights of the two child subtrees of any node differ by no more than one.

DF = Hight(Left Subtree) – Hight(Right Subtree)

Hight = the number of edges for the longest path from the node to the last leaf

Balance Factor (BF) = DF.

If ABS(BF) > 1 then it is not balanced.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

Node Is Balanced If...

- Should be -1, 0, or +1 for each node ➔ Abs(BF) <=1.
- If Abs(BF) > 1 then it is Imbalanced.
- If at any time they differ by more than one, rebalancing is done to restore this property.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

What is AVL?

- AVL trees, named after their inventors Adelson-Velsky and Landis, are self-balancing binary search trees.
- In an AVL tree, the heights of the two child subtrees of any node differ by no more than one.
- If at any time they differ by more than one, rebalancing is done to restore this property.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

In order to rebalance a node you need to do rotations

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

What are Rotations?

- To maintain balance after insertions and deletions, AVL trees use rotations.
- Rotations are pivotal tree manipulations that shift nodes and their subtrees around to move higher nodes lower and lower nodes higher, thus restoring the required balance of the tree.
- Concept of rotations as operations that reorder the nodes of the tree to maintain or restore balance.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience

Types of Rotations:

- There are four types of basic rotations:
  - RR – Right Rotation (Single Right Rotation)
  - LL – Left Rotation (Single Left Rotation)
  - LR – Left-Right (Double Rotations)
  - RL – Right-Left Rotations (Double Rotations)

Mohammed Abu-Hadhoud
MBA, PMOC, PgMPD, PMPD, PMI-RMPD, CM, ITILF, MCPD, MCSD
26+ years of experience


ProgrammingAdvices.com
Thank You
Mohammmed Abu-Hadhoud
26+ Years of Experience
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD
PROGRAMMING ADVICES
LEARN THE RIGHT WAY