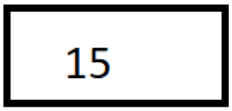
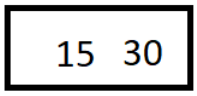
No 2 (B-Tree) – Chuunibyou Club

Insert 15



Tree kosong, buat node

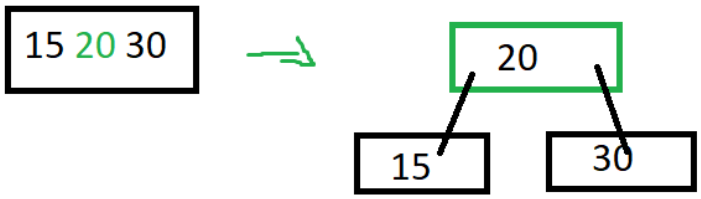
Insert 30



30 > 15, 30 kita tarok di kanan 15 karena sudah leaf

Karena ruangannya masih memenuhi kondisi key = m-1, tidak perlu di split

Insert 20



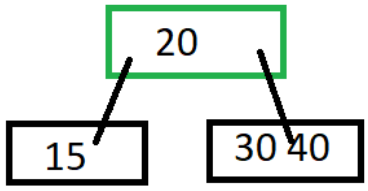
20 > 15 dan 20 < 30 , jdi 20 ditarok ditengah,

Jumlah key sudah lebih dari batas,

Median = 20, jdi 20 kita push ke atas

No 2 (B-Tree) – Chuunibyou Club

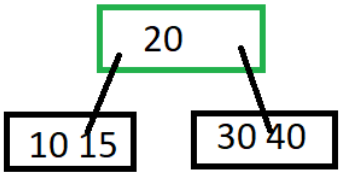
Insert 40



40 > 20, karena masih blm leaf, traverse ke kanan ketemu 30

40 > 30, masukan 40 ke kannanya 30, karena masih memenuhi kondisi B-tree tidak perlu split

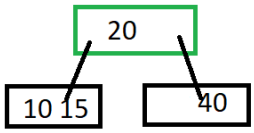
Insert 10



10 < 20, karena blm leaf, traverse ke kiri ketemu 15

10 < 15, masukan 10 ke kirinya 15, karena masih memenuhi kondisi B-tree tidak perlu split

Delete 30



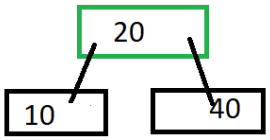
minimum key = 1
30 berada di node dgn key = 2
boleh langsung hapus

Traverse B-tree dari root u/ mencari 30

Saat ditemukan 30, nodenya memiliki jumlah key lebih dari minimum, sehingga 30 dapat dengan mudah dihapus & tidak perlu ngaapa-ngapain lagi

No 2 (B-Tree) – Chuunibyou Club

Delete 15

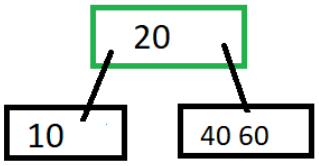


minimum key = 1
15berada di node dgn key = 2
boleh langsung hapus

Traverse B-tree dari root u/ mencari 15

Saat ditemukan 15, nodenya milikin jumlah key lebih dari minimum, sehingga 15 dapat langsung dihapus dan tidak perlu ngapa2ngapain lgi

Insert 60



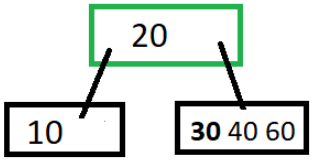
60 > 20, traverse ke kanan

60 > 40, masukan 60 ke kanannya 40, karena masih memenuhi kondisi B-tree tidak perlu di split

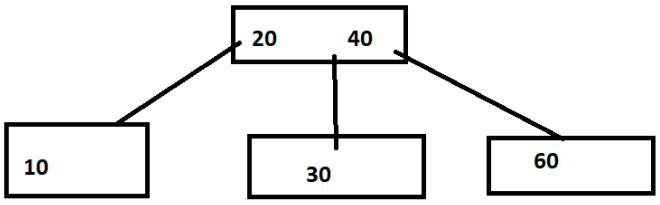
Insert 30

30 > 20, traverse ke kanan, ketemu 40

30 < 40, masukan 30 di kiri 40

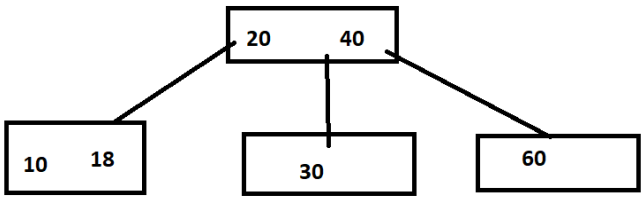


Karena dia lebih dari max (2) yaitu keynya sekarang ada 3 dlm 1 node, median harus naik jadi 40 naik ke atas, dan node yg berisi 30 dan 60 di split



No 2 (B-Tree) – Chuunibyou Club

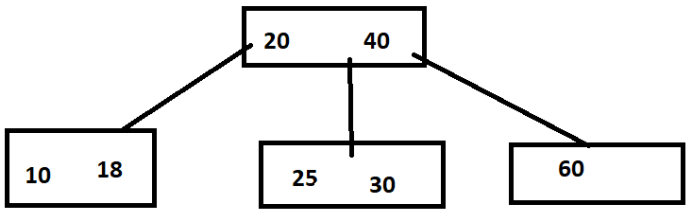
Insert 18



18 < 20, traverse ke kiri ketemu 10

18 > 10 , 18 masukan ke kanannya 10, masih memenuhi kondisi B-tree, tidak perlu di split

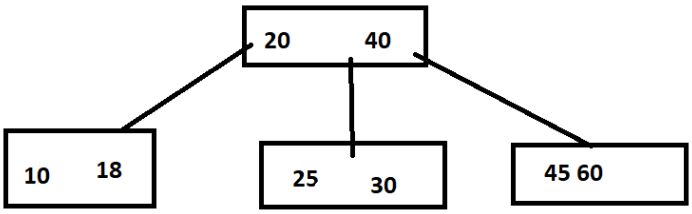
Insert 25



25 > 20, traverse ke kanan ketemu 30,

25 <30, 25 masukan ke kirinya 30, masih memenuhi kondsi B-tree, tidak perlu di split

Insert 45

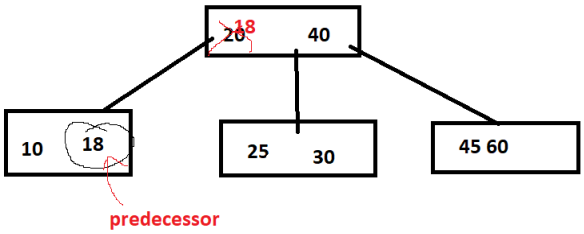


45 > 40, traverse ke kanan

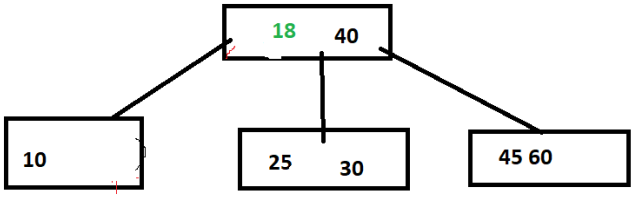
45 < 60, masukkan 45 ke kirinya 60, karena masih memenuhi kondisi B-tree, tidak perlu di split

No 2 (B-Tree) – Chuunibyou Club

Delete 20

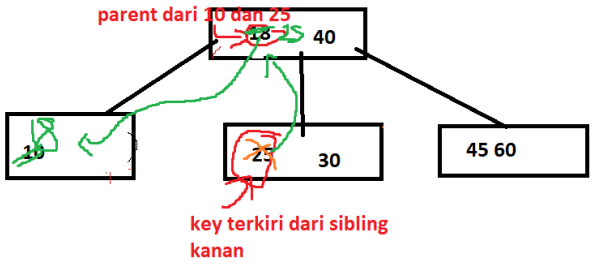


Search 20, kita temukan 20 ada di root, dan 20 merupakan internal



Sehingga kita mencari predecessor(ke kiri, lalu ke kanan terus hingga null) 20 yaitu 18, kita tukar 20 dengan 18, dan kita delete si predecessornya, saat ingin di delete predecessor berada di node yg lebih dari minimum, sehingga tinggal kita delete dan tidak perlu diapa-apakan

Delete 10



Search 10. 10 berada di leaf node,

Node tersebut merupakan node yg mengandung minimal key, jdi harus pinjam dari sibling

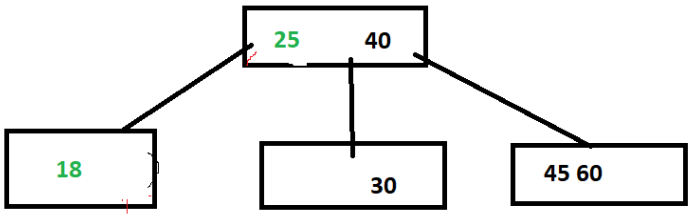
Cek sibling kiri, tidak ada

Cek sibling kanan ketemu yg lebih dari minimum yg berisi 25 dan 30, ambil key dari sibling kanan yang terkiri yaitu 25,

Parent dari 10 dan 25 yaitu 18 kita ubah menjadi 25

10 kita timpah dengan key dari parent yaitu 18, lalu 25 yang lama tinggal kita delete, karena dia lebih dari minimal

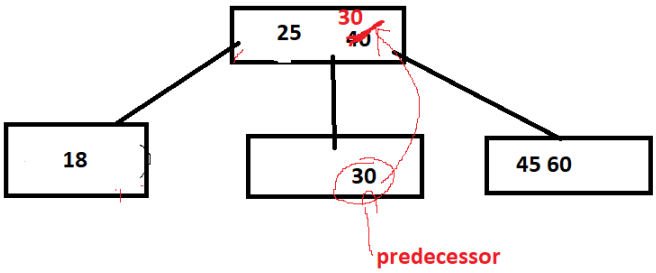
No 2 (B-Tree) – Chuunibyou Club



Delete 40

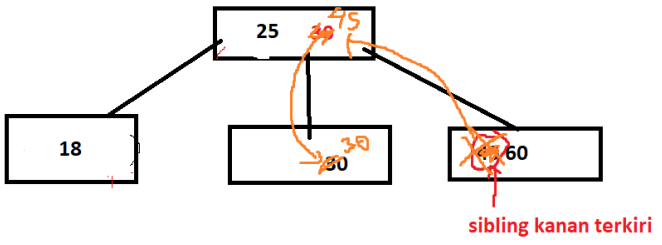
Search 40, 40 merupakan internal node, kit acari dlu predecessor dari 40 kita temukan 30

Kita ubah 40 dengan 30

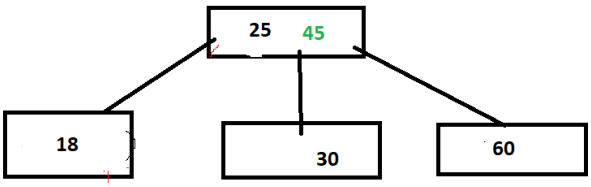


Lalu kita mau mendelete 30 yg lama, karena dia merupakana leaf node dengan minimum amount of key, kita harus meminjam dari sibling

Cek sibling kiri tidak ada

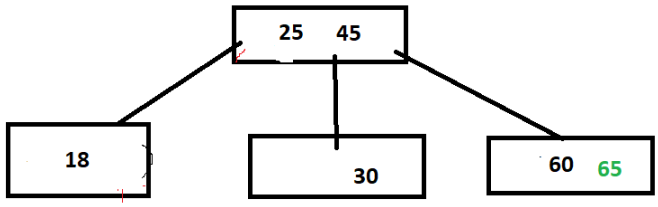


Cek sibling kanan ketemu 45 dan 65, dan mereka lebih dari minimum, karena sibling kanan kita ambil yg terkiri, kita dapat 45 dengan parent 45 dan 30 dulu yaitu 30 baru, 45 kita jadikan parent, dan parent dari 30 dan 45 yaitu 30 yang baru kita timpah saja ke 30



No 2 (B-Tree) – Chuunibyou Club

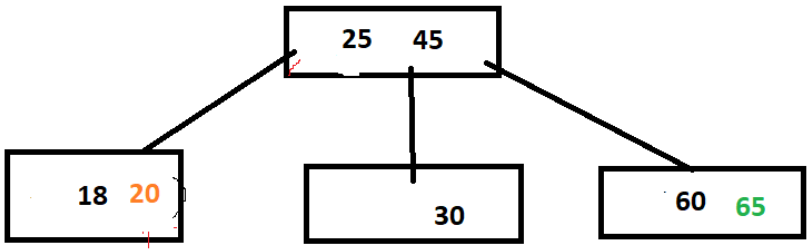
Insert 65



65 > 45, blm leaf, traverse ke kanan ketemu 60

65 > 60, masukan 65 ke kanannya 60, karena masih memenuhi B-tree tidak perlu di split

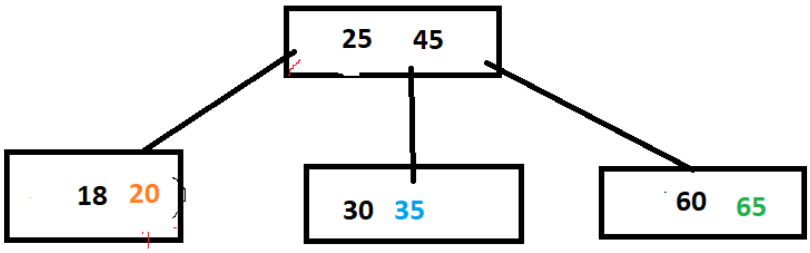
Insert 20



20 < 25,blm leaf, traverse ke kiri, ketemu 18

20 > 18 masukan 20 ke kanannya 18, karena masih memenuhi B-tree tidak perlu di split

Insert 35



35 > 25, blm leaf, traverse ke kanan, ketemu 30

35> 35, 35 masukan ke kanannya 30, karena masihi memenuhi B-tree tidak perlu di split