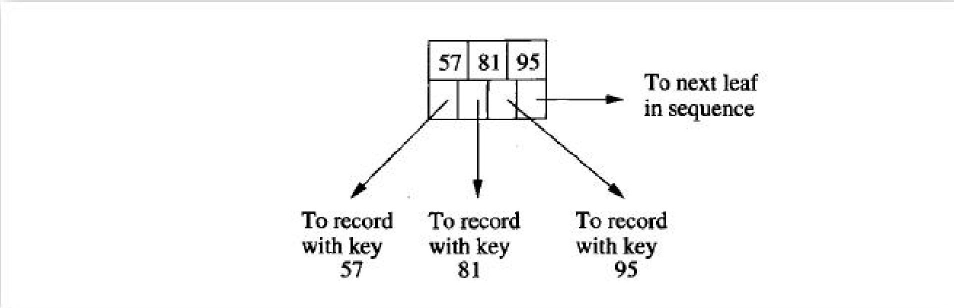
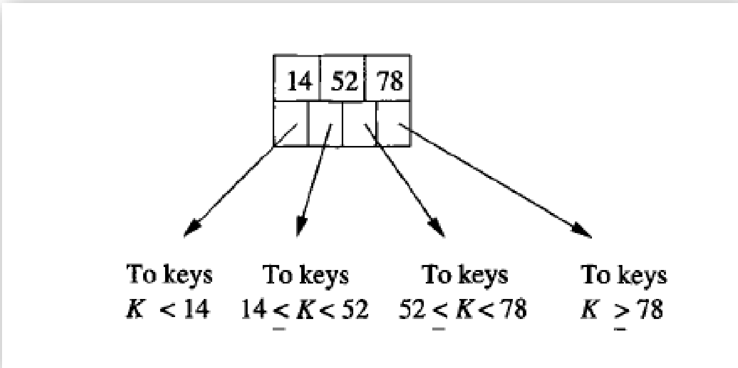
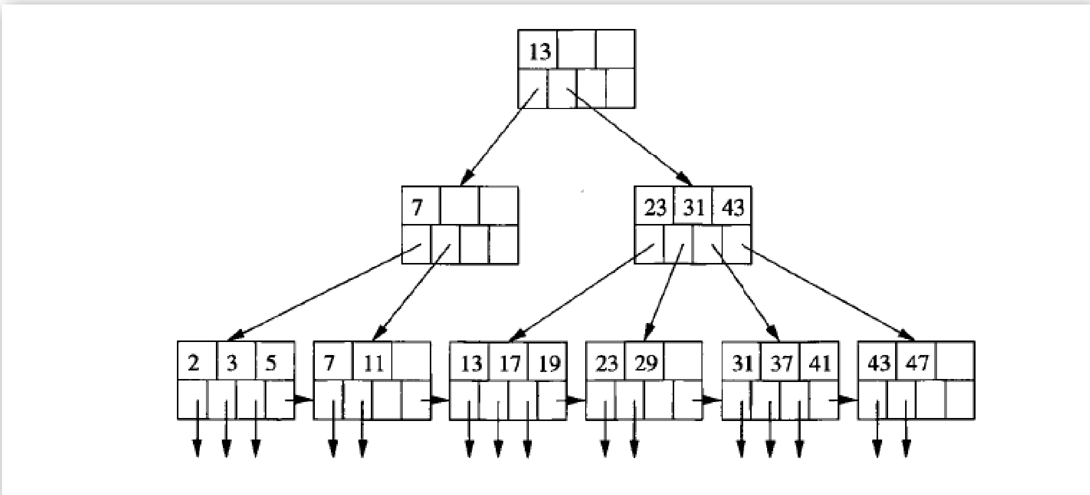



**4.21 ábra.** Egy B+-fa jellegzetes levele




**4.22 ábra.** Egy B+-fa jellegzetes belső csúcsa




**4.23 ábra.** B+-fa

### 4.3.3. Keresés B-fában

Térjünk most vissza az eredeti feltevésünkhöz, mely szerint a levelekben nincsenek ismétlődő kulcsok. Ez a feltevés megkönnyíti a B-fa műveleteinek tárgyalását, de nem feltétlenül szükséges a műveletekhez. Tegyük fel, hogy adott egy B-fa-index, és meg akarunk találni egy $K$ keresésikulcs-értékű rekordot. Rekurzív módon keressük a $K$-t, a gyökértől kezdünk, és egy levélnél fogunk megállni. A keresési eljárás a következő:

**Kiindulási pont:** Ha egy levélnél vagyunk, akkor végignézzük annak kulcsait. Ha az $i$-edik kulcs a $K$, akkor az *i*-edik mutató elvezet minket a keresett rekordhoz.

**Indukció:** Ha egy $K_1$, $K_2$, ..., $K_n$ kulcsokkal rendelkező belső csúcsnál vagyunk, akkor a 4.3.1. részben bemutatott szabályokat használjuk annak eldöntésére, hogy a csúcs melyik gyermekét vizsgáljuk meg a következőkben. Ez azt jelenti, hogy csak egy olyan gyermek van, amely elvezethet egy $K$ kulcsot tartalmazó levélhez. Ha $K < K_1$, akkor ez az első gyermek, ha $K_1 \le K < K_2$, akkor ez a második gyermek és így tovább. Az így megkapott gyermekre rekurzív módon alkalmazzuk a keresési szabályt.

**4.23. példa:** Tegyük fel, hogy adott a 4.23. ábrán látható B-fa, és szeretnénk találni egy olyan rekordot, amelynek keresési kulcsa 40. Elindulunk a gyökérből, ahol egyetlen kulcs van, a 13. Mivel $13 \le 40$, ezért a második mutatót követjük, amely a 23, 31 és 43 kulcsokkal rendelkező, második szinten található belső csúcshoz vezet bennünket.

Ennél a csúcsnál $31 \le 40 < 43$, így a harmadik mutatót követjük. Ily módon a 31, 37 és 41 kulcsokat tartalmazó levélhez jutunk. Ha lenne az adatfájlban olyan rekord, amelynek keresési kulcsa 40, akkor a 40-es kulcsot ebben a levélben találnánk. Mivel nem találtunk 40-es kulcsot, levonjuk a következtetést, miszerint az alapul szolgáló adatok nem tartalmaznak 40-es kulcsú rekordot.

Figyeljük meg, hogyha olyan rekordot kerestünk volna, amelynek kulcsa 37, akkor ugyanezeket a döntéseket hoztuk volna, de amikor eljutottunk volna a levélhez, megtaláltuk volna a 37-es kulcsot. Mivel ez a második kulcs a levélben, a második mutatót követve eljutunk a 37-es kulcsú adatrekordhoz. □

### 4.3.4. Tartományra vonatkozó lekérdezések

A B-fák nem csak olyan lekérdezések esetén hasznosak, amelyekben a keresési kulcs egy konkrét értékére keressük, hanem olyankor is, amikor értékek egy tartományára vonatkozik a kérdés. A *tartományra vonatkozó lekérdezések* a `WHERE` záradékban jellegzetesen tartalmaznak egy olyan kifejezést, amely az = és < > operátoroktól eltérő összehasonlító operátort tartalmaz. Példák *k* keresési kulcs attribútumot használó tartományt eredményező lekérdezésekre:

```
SELECT *
FROM R
WHERE R.k > 40;
```

vagy

```
SELECT *
FROM R
WHERE R.k >= 10 AND R.k <= 25;
```

Ha meg akarjuk találni egy B-fa leveleiben az összes $[a, b]$ tartományba tartozó kulcsot, akkor végrehajtunk egy keresést az $a$ megtalálására. Függetlenül attól, hogy létezik-e vagy sem, eljutunk egy olyan levélhez, ahol az $a$ előfordulhatna, és megkeressük a levélben azokat a kulcsokat, amelyek nagyobbak vagy egyenlők, mint az $a$. Minden ilyen kulcshoz találunk egy mutatót, amely egy olyan rekordra mutat, amelynek kulcsa a kívánt tartományba tartozik.

Ha nem találunk olyan kulcsot, amely nagyobb, mint $b$, akkor használjuk a levélnek azt a mutatóját, amely a következő levélre mutat. Megtartjuk a megvizsgált kulcsokat, valamint követjük a hozzájuk tartozó mutatókat, mindaddig, amíg:

1. Találunk egy olyan kulcsot, amely nagyobb, mint $b$, és ekkor megállunk.
2. Elérjük a levél végét, ekkor továbblépünk a következő levélre, és megismételjük az eljárást.

A fenti keresési algoritmus akkor is működik, ha $b$ végtelen, azaz csak egy alsó határ van megadva, felső határ nincs. Ebben az esetben végignézzük az összes levelet attól a levéltől kezdve, amelyik tartalmazhatná az $a$ kulcsot, egészen a levelek végéig. Ha az $a$ értéke $-\infty$ (azaz a tartománynak csak felső határa van, alsó határa nincs), akkor a „mínusz végtelen” kulcs keresése a B-fa valamennyi csúcsa esetén az első gyermekhez vezet majd bennünket, azaz tulajdonképpen az első levelet találjuk majd meg. A keresés a továbbiakban ugyanúgy történik, mint fentebb, megállni akkor kell majd, ha túlléptük a $b$ kulcsot.

**4.24. példa:** Tegyük fel, hogy adott a 4.23. ábrán látható B-fa, és a (10, 25) tartományba eső kulcsokat keressük. Elkezdjük a 10-es kulcs keresését, és eljutunk a második levélhez. Az első kulcs kisebb, mint 10, de a második 11, ami nagyobb vagy egyenlő, mint 10. Követjük a hozzá tartozó mutatót, hogy megkapjuk a 11-es kulcsú rekordot.

Mivel nincs több kulcs a második levélben, követjük a levelek láncolatát, és eljutunk a harmadik levélhez, melynek kulcsai 13, 17 és 19. Mindegyik kisebb vagy egyenlő, mint 25, ezért követjük a hozzájuk tartozó mutatókat, és megkapjuk azokat a rekordokat, amelyek ezekkel a kulcsértékekkel rendelkeznek. Végezetül átmegyünk a negyedik levélbe, ahol először 23-as kulcsot találunk. A levél következő kulcsa azonban 29, ami nagyobb, mint 25, ezért itt be is fejezzük a keresést. Ily módon megkaptuk azt az öt rekordot, melynek kulcsai 11, 13, 17, 19 és 23. □

### 4.3.5. Beszúrás B-fában

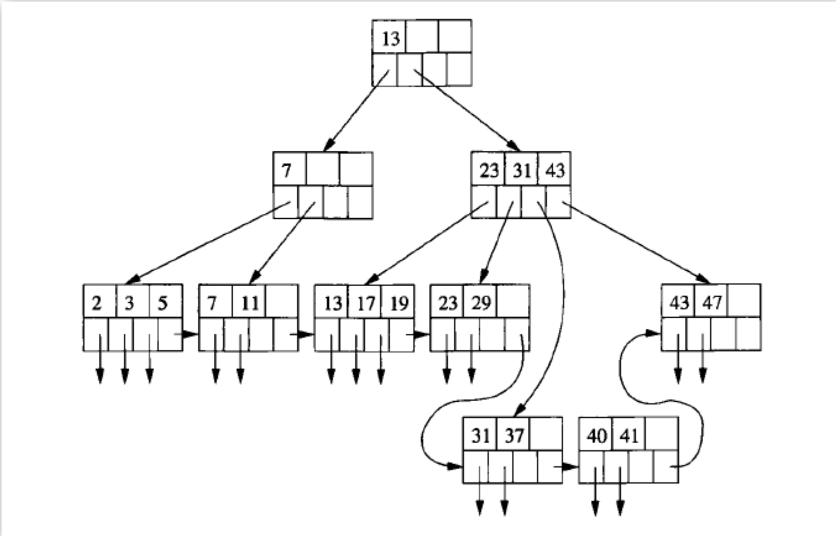
A B-fáknak vannak előnyei az egyszerűbb többszintű indexekkel szemben, ezek közül láthatunk néhányat, miközben áttekintjük, hogy miként kell beszúrni egy B-fába egy új kulcsot. A megfelelő rekordot a 4.1. részben bemutatott módszerek valamelyikével beszúrjuk a B-fával indexelt fájlba; itt most azt tekintjük át, hogy a B-fa ennek megfelelően miként változik. A beszúrás alapjában véve rekurzív:

- Megpróbálunk találni egy helyet az új kulcs számára a megfelelő levélben, és ha van szabad hely, akkor ide tesszük.
- Ha nincs hely a megfelelő levélben, akkor kettévágjuk a levelet, és szétosztjuk a kulcsokat a két új csúcs között, így mindkettő félig lesz telítve vagy éppen csak egy kissé jobban.
- Egy csúcs szétvágása egy adott szinten hatással van a fölötte levő szintre is, oly módon, hogy egy új kulcs-mutató párt kell beszúrni ezen a felsőbb szinten. Ily módon rekurzívan alkalmazhatjuk ezt a stratégiát a magasabb szinten történő beszúrásra: ha van hely, beszúrjuk amit kell; ha nincs, akkor szétvágjuk a szülő csúcsot és megyünk tovább fölfelé a fában.
- Van egy kivétel: ha a gyökérbe próbálunk beszúrni és nincs hely, akkor szétvágjuk a gyökeret két csúcsra, és létrehozunk egy új gyökeret a következő szinten; az új gyökérnek a szétvágás következtében két gyermek csúcsa lesz. Emlékezzünk vissza, hogy bármekkora is az $n$ (az egy csúcsba tehető kulcsoknak fenntartott helyek száma), a gyökér számára mindig engedélyezett, hogy csak egy kulcsa és két gyermeke legyen.

Amikor szétvágunk egy csúcsot, és beszúrunk a szülő csúcsba, vigyáznunk kell arra, hogy miként kezeljük a kulcsokat. Először is, tegyük fel, hogy az $N$ egy olyan levél, amelynek kapacitása $n$ kulcs. Tegyük fel továbbá, hogy szeretnénk beszúrni egy $(n + 1)$-edik kulcsot és a hozzá tartozó mutatót. Készítünk egy új $M$ csúcsot, amely az $N$ testvére lesz, közvetlenül jobbra tőle. Az első $\left\lceil \frac{n+1}{2} \right\rceil$ kulcs-mutató pár a kulcsok rendezett sorrendjében az $N$ csúcsban marad, míg a többi kulcs-mutató pár átköltözik az $M$ csúcsba. Figyeljük meg, hogy az $M$ és az $N$ csúcs egyaránt elegendő számú kulcs-mutató párral rendelkezik, legkevesebb $\left\lfloor \frac{n+1}{2} \right\rfloor$ párral.

Most tegyük fel, hogy az $N$ egy olyan belső csúcs, melynek kapacitása $n$ kulcs és $n + 1$ mutató, de az $N$ csúcshoz $n + 2$ mutató kellene tartozzon egy csúcs alsóbb szinten történt szétvágása miatt. A következőket tesszük:

1. Készítünk egy új $M$ csúcsot, amely az $N$ testvére lesz, közvetlenül jobbra tőle.
2. Az első $\left\lceil \frac{n+2}{2} \right\rceil$ mutató, a kulcsok rendezett sorrendjében az $N$ csúcsban marad, míg a többi $\left\lfloor \frac{n+2}{2} \right\rfloor$ átköltözik az $M$ csúcsba.
3. Az első $\left\lceil \frac{n}{2} \right\rceil$ kulcs az $N$ csúcsban marad, míg a többi $\left\lfloor \frac{n}{2} \right\rfloor$ átköltözik az $M$ csúcsba. Figyeljük meg, hogy középen marad egy kulcs, amely nem jelenik sem az $N$, sem az $M$ csúcsban. A maradék $K$ kulcs azt a legkisebb kulcsot jelöli, amely az $M$ első gyermekén keresztül elérhető. Habár ez a kulcs nem jelenik meg sem az $N$, sem az $M$ csúcsban, mindamellett az $M$ csúcshoz tartozik abban az értelemben, hogy az $M$ csúcson keresztül elérhető legkisebb kulcsot jelöli. Ekképpen a $K$-t az $N$ és $M$ csúcsokhoz tartozó szülő fogja használni, hogy megossza a kereséseket a két csúcs között.




**4.25 ábra.** A 40-es kulcs beszúrásának kezdete

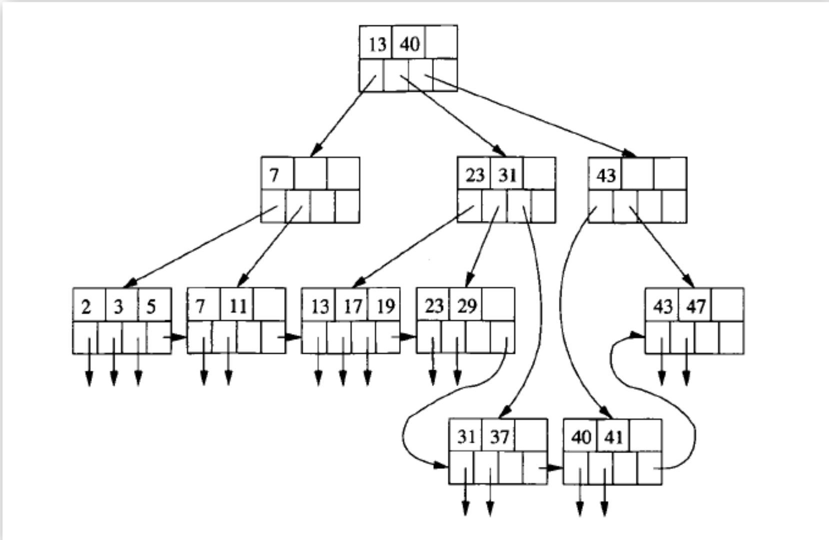
**4.25. példa:** Szúrjuk be a 4.23. ábrán látható B-fába a 40-es kulcsot. A beszúráshoz megfelelő levelet a 4.3.3. részben leírt eljárással keressük meg. Ahogyan a 4.23. példában is láttuk, a beszúrás az ötödik levélbe történik. Mivel $n = 3$, és ez a levél most négy kulcs-mutató párt tartalmaz – 31, 37, 40 és 41 – szét kell vágnunk a levelet. Az első lépés az, hogy készítünk egy új csúcsot, és a két legnagyobb kulcsot (40 és 41) áttesszük ebbe az új csúcsba. A 4.25. ábrán láthatjuk ezt a szétvágást.

Megjegyzendő, hogy habár most négy sorban ábrázoljuk a csúcsokat, a fának valójában három szintje van, és a hét levél foglalja el az ábra két alsó sorát. A levelek össze vannak kötve az utolsó mutatóik segítségével, amelyek most is egy balról jobbra tartó láncot alkotnak.

Be kell most szúrnunk egy új mutatót az új levélhez (ahhoz, amelynek kulcsai a 40 és a 41) a fölötte levő csúcsba (amelynek kulcsai 23, 31 és 43). Ehhez a mutatóhoz társítanunk kell a 40-es kulcsot, amely az új levélen keresztül elérhető legkisebb kulcs. Sajnos, a szétvágott csúcs szülője is tele van; nincs benne hely egy újabb kulcsnak vagy mutatónak. Ily módon ezt is szét kell vágnunk.

Kezdjük azokkal a mutatókkal, amelyek az utolsó öt levélre mutatnak, és a négy utolsó levél legkisebb kulcsainak a listájával. Tehát adottak a $P_1$, $P_2$, $P_3$, $P_4$, $P_5$ mutatók azokhoz a levelekhez, amelyeknek legkisebb kulcsai 13, 23, 31, 40 és 43, és adott egy, a mutatók elválasztására szolgáló kulcssorozatunk: 23, 31, 40, 43. Az első három mutató és az első két kulcs a szétvágott belső csúcsban marad, míg az utolsó két mutató és az utolsó kulcs átmegy az új csúcsba. A megmaradt kulcs, a 40-es, az új csúcson keresztül elérhető legkisebb kulcsot jelöli.

A 4.26. ábra a 40-es kulcs beszúrásának befejezését mutatja be. A gyökérnek most három gyermeke van; a két utolsó a szétszedett belső csúcsból származik. Figyeljük meg, hogy a 40-es kulcs, amely a szétszedett csúcsok második csúcsán keresztül elérhető legkisebb kulcs, a gyökérben került elhelyezésre, hogy szétválassza a gyökér második és harmadik gyermekeinek a kulcsait. □




**4.26 ábra.** A 40-es kulcs beszúrásának befejezése