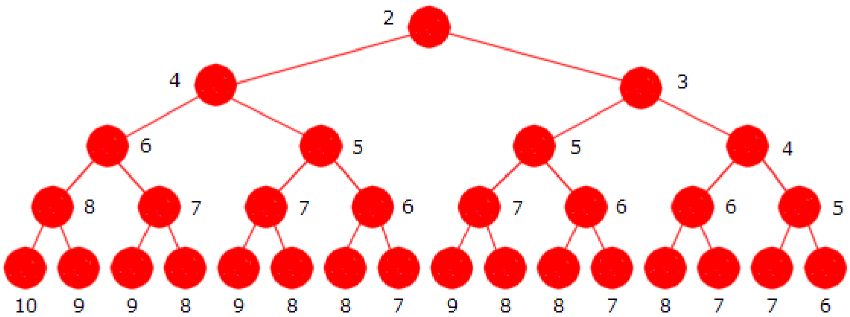
❑ Binærsøk – antall sammenligninger

2. versjon av binærsøk (se *Programkode* 1.3.6 *b*) inneholder flg. kode:

```
if (verdi > midtverdi) v = m + 1;
else if (verdi < midtverdi) h = m - 1;
else return m;
```

Figur 1 nedenfor inneholder et beslutningstre for denne versjonen av binærsøk. Det er det samme treet som i *Figur* 1.3.7 *e)* bortsett fra at verdiene i nodene er tatt vekk. Her er vi kun interessert i antallet sammenligninger som skal til for å finne den verdien som ligger i en bestemt node. Dette antallet er skrevet opp ved siden av hver node:



Figur 1: Et beslutningstre for 2. versjon av binærsøk

Vi ser både fra figuren og fra koden øverst, at det trengs én sammenligning mer for å finne verdien i en nodes høyre barn enn for å finne verdien i noden selv. Det trengs to sammenligninger mer for å finne verdien i en nodes venstre barn enn for å finne verdien i noden selv.

Vi antar at 2. versjon av binærsøk brukes på en tabell med lengde $n = 2^k - 1$ der $k = 1, 2, 3, \ldots$ Vi får treet i *Figur* 1 når $n = 2^5 - 1 = 31$. Vi ser at treet har 5 nivåer, dvs. fra nivå 0 til nivå 4. Hvis tabellen har lengde $n = 2^k - 1$, vil binærsøk gi et perfekt binærtre som beslutningstre. Treet vil ha k nivåer og n noder.

Vi skal finne en formel A_k for det sammenlagte antallet sammenligninger som utføres når hver eneste verdi skal finnes. I *Avsnitt* 1.3.7 kom vi frem til formelen

(1) $$A_k = (\frac{3}{2}k - 1) \cdot 2^k + 1$$

ved å observere at for hvert nivå i binærtreet var summen av antallene for første og siste node det samme som summen for andre og nest siste node, osv. Dette ble brukt til å sette opp potensrekken:

(2) $$A_k = \frac{4}{2} \cdot 2^0 + \frac{7}{2} 2^1 + \frac{10}{2} 2^2 + \frac{13}{2} 2^3 + \ldots + \frac{3k+1}{2} 2^{k-1}$$

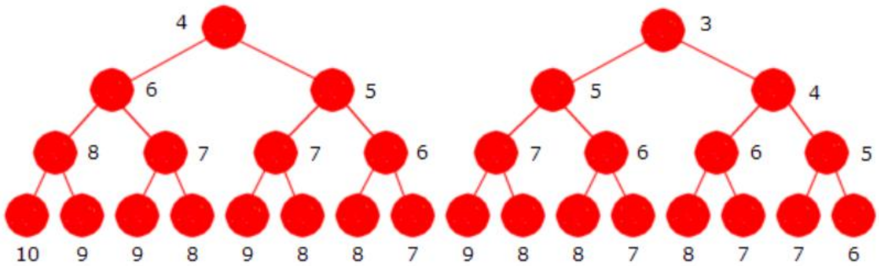
Dette kan skrives som

(3) $\quad A_k = \sum_{i=0}^{k-1}(\frac{3}{2}i+2)\cdot 2^i \;=\; \frac{3}{2}\sum_{i=0}^{k-1} i\cdot 2^i \;+\; 2\sum_{i=0}^{k-1} 2^i$

Ved å bruke formlene G.1.9 og G.1.7 fra formelsamlingen i *Vedlegg* G får vi at summen av potensrekken i (3) blir lik:

(4) $\quad A_k = \frac{3}{2}[(k-2)\cdot 2^k + 2] \;+\; 2\cdot(2^k - 1) \;=\; (\frac{3}{2}k - 1)\cdot 2^k + 1$

Vi skal nå finne formelen ved hjelp av en differensligning. Rotnoden i *Figur* 1 har to subtrær med samme form som hele treet, men subtrærne har et antall nivåer som er én mindre enn treet. De to subtrærne er satt opp i *Figur* 2 nedenfor:



Figur 2: De to subtrærne til rotnoden i *Figur* 1

Hvis vi fjerner nederste rad i treet i *Figur* 1 og trekker fra to i antallet ved hver node i det venstre subtreet, får vi to like trær. Vi får også det samme treet hvis vi trekker fra én i antallet ved hver node i det høyre subtreet. Med $n = 2^k - 1$ noder i hele treet blir det $n = 2^{k-1} - 1$ noder i subtrærne. Det gir differensligningen:

(5) $\quad A_k \;=\; A_{k-1} \;+\; 2(2^{k-1} - 1) \;+\; A_{k-1} \;+\; (2^{k-1} - 1) \;+\; 2$

Dette kan omformes til:

(6) $\quad A_k \;=\; 2A_{k-1} \;+\; f(k) \quad$ der $\; f(k) = 3\cdot 2^{k-1} - 1$

Den homogene delen av differensligningen (6) har generell løsning lik:

(7) $\quad A_k \;=\; \alpha 2^k$

Siden vi har 2 opphøyd i en eksponent både i den generelle løsningen (7) og i leddet $f(k)$ på høyre side i (6), må den inhomogene differensligningen (6) ha en partikulær løsning på formen:

(8) $\quad A_k \;=\; \beta k 2^{k-1} + \gamma$

Ved å sette inn i (6) finner vi at $\beta = 3$ og $\gamma = 1$. Det betyr at den generelle løsningen av (6) er:

(9) $\quad A_k \;=\; \alpha 2^k \;+\; 3k\cdot 2^{k-1} + 1$

Ved å bruke $A_1 = 2$ får vi $\alpha = -1$. Dermed blir formelen for A_k lik:

$$A_k = -2^k + 3k \cdot 2^{k-1} + 1 = (\frac{3}{2}k - 1)2^k + 1 \qquad (10)$$