4.1: Induktionsbevis.

For $n \in \mathbb{Z}$, $n \geq 1$, lad $P(n)$ betegne et udsagn, der kan være sandt eller falsk. Sandhedsværdien kan være forskellig for forskellige værdier af n.

Induktionsprincippet: For at bevise at $P(n)$ er sand for alle $n \geq 1$ skal vi
Basisskridt: bevise at $P(1)$ er sand.
Induktionsskridt: bevise at der for ethvert $k \geq 1$ gælder: hvis $P(k)$ er sand så er $P(k+1)$ også sand.

Man kan eventuelt ændre alle røde 1-taller til et andet tal b. Det grønne 1-tal må ikke ændres.

Procedure Bubblesort ($a_1, \ldots, a_n$: tal, $n \geq 2$)

for $i := 1$ **to** $n - 1$

begin

    for $j := 1$ **to** $n - i$

    begin

        if $a_j > a_{j+1}$ **then** ombyt a_j og a_{j+1}

        { a_{j+1} er det største af tallene $a_1, \ldots, a_{j+1}$ }

    end

    { $a_{n+1-i}, \ldots, a_n$ står på de rigtige pladser }

end

{ $a_1, \ldots, a_n$ er sorteret i ikke-aftagende rækkefølge}

4.2: Stærk induktion.

For $n \in \mathbb{Z}$, $n \geq 1$, lad $P(n)$ betegne et udsagn.

Induktionsprincippet: For at bevise at $P(n)$ er sand for alle $n \geq 1$ skal vi

Basisskridt: bevise at $P(1)$ er sand.

Induktionsskridt: bevise at der for ethvert $k \geq 1$ gælder: hvis $P(1), \ldots, P(k)$ alle er sande så er $P(k+1)$ også sand.

Man kan eventuelt ændre alle røde 1-taller til et andet tal b. Det grønne 1-tal må ikke ændres.

4.2: Velordningsprincippet.

Lad S være en *ikke-tom* mængde af ikke-negative hele tal.
Så har S et mindste element, altså et element $m \in S$ så $s \geq m$ for alle $s \in S$.

Anvendelse: $P(n)$ et udsagn. Vi skal bevise at $P(n)$ er sand for alle $n \geq 0$.
Lad $S = \{n \text{ ikke-negativ heltal} \mid P(n) \text{ er falsk}\}$.
Skal vise: $S = \emptyset$.
Bevis ved modstrid: antag $S \neq \emptyset$ og lad $m \in S$ være det mindste element. ...

4.4: Rekursive algoritmer.

"Algoritme 3": Rekursiv beregning af a^n Side 313

procedure power (a reelt tal $\neq$ 0, n: ikke-negativt heltal)

if $n = 0$ **then** power$(a, n) = 1$

else if n lige **then** power$(a, n) =$ power$(a, \frac{n}{2})^2$

else power$(a, n) = a \cdot$ power$(a, \frac{n-1}{2})^2$

procedure mergesort($L = a_1, \ldots, a_n$ liste af tal)
if $n > 1$ **then**
begin

$m := \lfloor \frac{n}{2} \rfloor$

$L_1 := a_1, \ldots, a_m$

$L_2 := a_{m+1}, \ldots, a_n$

$L :=$ merge(mergesort(L_1), mergesort(L_2))

end

{ L er sorteret i ikke-aftagende rækkefølge}

Kompleksitet: hvor mange sammenligninger skal man foretage for at sortere en liste med n elementer ved hjælp af Mergesort?

Lemma 1. For at flette to sorterede lister med n og m elementer bruges højst $m + n - 1$ sammenligninger.

Sætning 1. Antal sammenligninger, der i alt bruges af Mergesort for at sortere en liste med $n = 2^k$ elementer med er højst

$$2^k(k - 1) + 1 = n(\log(n) - 1) + 1.$$

Mergesort har kompleksitet $O(n \log(n))$.