

3/19 algorithm

do
 1 $B_1 \rightarrow SL_1$
 1 $B_2 \rightarrow SL_2$
 :
 1 $B_n \rightarrow SL_n$
 od

if
 } [alternative structure
 } [repetitive " "
 fi

if $B \rightarrow S_A$
 1 $\neg B \rightarrow S_B$
 fi
 ↑
 if B then S_A [else S_B]
 while B do S

~~*~~ for $i = I$ to F do of
od

$i := I$;
 do $i \leq F \rightarrow$ intitidize
 → test
 do $i := i + 1$ → update
 od

if do
 $B_1 \rightarrow$
 1 $B_2 \rightarrow$ "
 :
 1 $B_n \rightarrow$
 fi od

if fi do od
 \equiv abort \equiv skip

$max := a_1; i := 2$
 do $i \leq n \rightarrow$ if $max \leq a_n \rightarrow max := a_n; i := i + 1$
 1 $max \geq a_n \rightarrow$ ~~max~~ $i := i + 1$;
 fi
 od

$max := a_1$;
 for $i := 2$ to n do if $max \leq a_i := max := a_i$
 (else skip)
 od fi

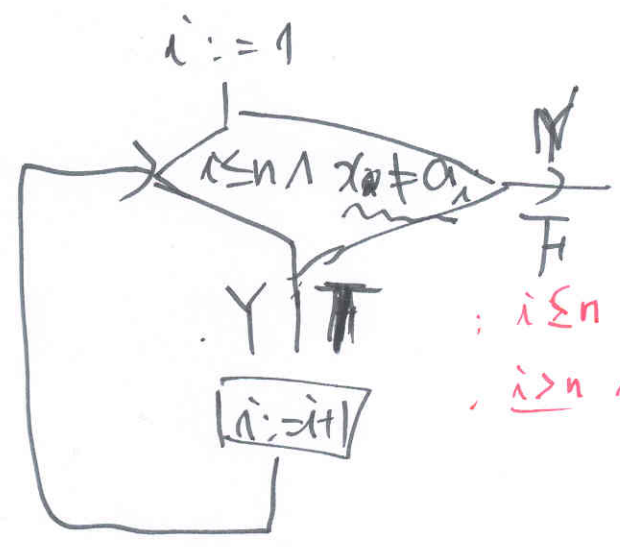
\rightarrow
 $\forall i : 1 \leq i \leq n : max \geq a_i$

$\equiv (max \geq a_1) \wedge (max \geq a_2) \wedge \dots \wedge (max \geq a_n)$

alg. 2 $\rightarrow a_1$ $x = a_i$



a_n



$i \leq n \wedge x = a_i$
 $i > n \wedge x \neq a_i$

linear search $n \frac{n}{2}$

binary search $\log_2 n$ — sorted!