

Exercise 1: Analyzing an Algorithm

a) A suitable loop invariant \mathcal{INV} is:

- (i) $A[i, \dots, n]$ is sorted non-descended
- (ii) $\forall k, l \in \mathbb{N} : 1 \leq k < i \wedge i \leq j \leq n \implies A[k] \leq A[l]$

b) Prove through Induction:

base case: $i = n$

- (i) $A[i, \dots, n] = A[n, \dots, n] = A[n]$, an one-element array is always sorted
- (ii) $\forall k, l \in \mathbb{N} : 1 \leq k < i \wedge i \leq l \leq n : A[k] \leq A[l]$
 $\iff \forall k, l \in \mathbb{N} : 1 \leq k < n \wedge n \leq j \leq n \implies A[k] \leq A[l]$. Let $A[m]$ be the greatest element in A , that means $A[(m+1)-1] \geq A[m+1]$, so it would be moved to the next index, after increasing i, j, m , by one this would repeat until $j = n$ and $A[m]$ was moved to the last index.

induction step: $i \rightsquigarrow i - 1$

induction hypothesis

- (i) $A[i, \dots, n]$ is sorted non-descended
- (ii) $\forall k, l \in \mathbb{N} : 1 \leq k < i \wedge i \leq j \leq n \implies A[k] \leq A[j]$

Because of the induction hypothesis we now that $A[i, \dots, n]$ is sorted non-descending (i) and that $A[i-1] \leq A[i]$ (ii), so $A[i-1, \dots, n]$ is sorted non-descending. Let $A[m]$ be the greatest element in $A[1, \dots, i]$, that means $A[(m+1)-1] \geq A[m+1]$, so it would be moved to the next index, after increasing i, j, m , by one this would repeat until $j = i$ and $A[m]$ was moved to the index i . So that $\forall k \in \mathbb{N} : 1 \leq k < i : A[k] \leq A[i]$. And because we know that $\forall k, l \in \mathbb{N} : 1 \leq k < i \wedge i \leq j \leq n \implies A[k] \leq A[j]$, because of the induction hypothesis, we have now proven (i) and (ii). ■

c) (i) We have proven, that \mathcal{INV} is a loop invariant.

(ii) In the beginning the Array is unsorted, but the sorted area is empty, so the \mathcal{INV} is true

(iii) after the $n-1$ th iteration the Array $A[1, \dots, n]$ is sorted non-descending because the Array $A[2, \dots, n]$ is sorted non-descending and the element $A[1]$ is not greater than every element in the Array $A[2, \dots, n]$.

(iv) the loop terminates after $n - 1$ iterations

Exercise 2: Hoar-Logic - Analysis

a) $x - 2 < 0 \iff x < 2$

b) $z - 5 > 5 \iff z > 10$

Exercise 3: Dynamic Programming - Knapsack

| B | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|-------------------------------|---|---|---|---|---|---|----|
| I_1 | 0 | 2 | 4 | 6 | 7 | 9 | 11 |
| I_2 | 0 | 2 | 4 | 6 | 7 | 9 | 11 |
| I_3 | 0 | 1 | 4 | 5 | 7 | 9 | 10 |
| I_4 | 0 | 1 | 4 | 5 | 7 | 9 | 10 |
| <i>using only other items</i> | 0 | 1 | 4 | 5 | 7 | 8 | 9 |