Exercise Sheet 7

Issue Date: December 5th, 2023
Due Date: December 11th, 2023 – 10:00 a.m.
$\sum$ 10 Points

Konzepte der Informatik INF-11700
Winter 2023/2024

Universität Konstanz

University of Konstanz
Dr. Barbara Pampel
Sabrina Jaeger-Honz

Complexity, Verification & Correctness

Exercise 1: Analyzing an Algorithm (*6 points*)

Given the following algorithm working on an array $A[1, \ldots, n]$:

```
1 f(A[1,...,n]) begin
2 |  for i ← n, n − 1, ..., 2 do
3 |  |  for j ← 2, 3, ..., i do
4 |  |  |  if A[j − 1] ≥ A[j] then
5 |  |  |  |  swap value in A[j − 1] and A[j]
```

a) (2 points) Give a suitable loop invariant $\mathcal{INV}$ that can be used in Floyd's Verification Method.

b) (2 points) Prove the loop-invariant by induction.

c) (2 points) Prove the correctness of algorithm by using Floyd's verification method.

Exercise 2: Hoar-Logic – Analysis (*2 points*)

Give the weakest precondition for the following assignments and conditions:

a) $\mathbf{WP}[\![\texttt{x = x - 2;}]\!](x < 0)$

b) $\mathbf{WP}[\![\texttt{x = z - 5;}]\!](x > 5)$

Exercise 3: Dynamic Programming – Knapsack (*2 points*)

You have backpacks $B_0, \ldots, B_6$ which can be filled up to a maximum weight j for a backpack B_j. You have items $I_1, \ldots, I_n$ with values $v(I)$ and weights $w(I)$. Four of these are:

$$I_1 : v(1) = 3, w(1) = 2$$
$$I_2 : v(2) = 2, w(2) = 1$$
$$I_3 : v(3) = 1, w(3) = 2$$
$$I_4 : v(4) = 5, w(4) = 3$$

About the *other items* you don't know anything other than the maximal values using only these items (given in the last row). Use the *Knapsack* algorithm from the lecture on Dynamic

Programming to update the table below. In there, cell $[i, j]$ records the maximum value that a backpack B_j with weight-limit j can carry when being allowed to pack items $I_i \ldots I_n$.

B	0	1	2	3	4	5	6
I_1							
I_2							
I_3							
I_4							
using only other items	0	1	4	5	7	8	9