Universität Augsburg
Institut für Informatik

Übung zur Vorlesung Informatik 1 WS 2017/18
Fakultät für Angewandte Informatik
Lehrprofessur für Informatik
Prof. Dr. Lorenz, Marius Brendle, Johannes Metzger, Lev Sorokin 17.01.2018

Übungsblatt 11

Abgabe: 24.01.2018, 12:00 Uhr (Postkasten der Veranstaltung und E-Mail an Tutor)

- Dieses Übungsblatt muss im Team abgegeben werden (Einzelabgaben sind nicht erlaubt).
- Bitte zur Angabe von Namen, Übungsgruppe und Teamnummer das **Deckblatt** verwenden!
- Die **Zeitangaben** geben zur Orientierung an, wie viel Zeit für eine Aufgabe später in der Klausur vorgesehen wäre; gehen Sie davon aus, dass Sie zum jetzigen Zeitpunkt wesentlich länger brauchen und die angegebene Zeit erst nach ausreichender Übung erreichen.

* leichte Aufgabe / ** mittelschwere Aufgabe / *** schwere Aufgabe

Aufgabe 41 ** *(Rekursion, 15 Minuten)*

a) (**, 4 Minuten)
Betrachten Sie die folgende **induktive Definition** einer Funktion *exp*:

$$exp(n) = \begin{cases} 1 & , \text{ falls } n = 0 \\ 2 \cdot exp(n-1) & , \text{ falls } n > 0 \end{cases}$$

Implementieren Sie eine **rekursive C-Funktion** `int exp(int n)`, die bei Übergabe einer nicht-negativen ganzen Zahl `n` den Wert von `exp(n)` berechnet und zurückgibt.

b) (**, 6 Minuten)
Betrachten Sie den folgenden in Pseudocode gegebenen **endständig rekursiven Algorithmus**:

Algorithmus : A
Eingabe : $a \in \mathbb{N}, b \in \mathbb{N}$
wenn $b = 1$ **dann**
  Ausgabe : a
sonst
  Ausgabe : $A(a \cdot a, b-1)$

- Welche mathematische Funktion berechnet der Algorithmus?
- Geben Sie den Algorithmus in *iterativer* Form als Programmablaufplan an.

c) (***, 5 Minuten)
Entwerfen Sie einen *rekursiven Algorithmus in Form von Pseudocode*, der bei Eingabe einer Folge ganzer Zahlen $a_1, \ldots, a_n$ ($n \in \mathbb{N}, n > 0$) das Maximum der Folge berechnet und zurückgibt.

Hinweis: Führen Sie die Berechnung des Maximums der Folge $a_1, \ldots, a_n$ auf die Berechnung des Maximums der Folge $a_1, \ldots, a_{n-1}$ zurück.

Aufgabe 42 ** *(Prädikatenlogik, 16 Minuten)*

In den folgenden Aufgaben seien x, y Variablen über der Grundmenge $\mathbb{Z}$ und z eine Variable über der Grundmenge $\mathbb{R}$.

a) (*, 5 Minuten)
Entscheiden Sie, ob im Folgenden jeweils prädikatenlogische Formeln angegeben sind oder nicht (ohne Angabe einer Begründung, jeweils 1 Minute):

1. $\exists x(\forall y)$
2. $y > 0$
3. $\exists x(x + 3)$
4. $\forall y(y \neq 0)$
5. $x \leq \forall y$

b) (**, 5 Minuten)
Geben Sie an, ob die folgenden prädikatenlogischen Formeln erfüllt sind oder nicht (ohne Angabe einer Begründung, jeweils 1 Minute).

1. $\forall x(\exists y(x = y))$
2. $\exists y(\forall x(x = y))$
3. $\forall x((x > 0) \vee (x < 0))$
4. $\forall x((x \mod 4 = 0) \implies (x \mod 2 = 0))$
5. $(\forall x(x \mod 4 = 0)) \implies (\forall x(x \mod 2 = 0))$

c) (**, 3 Minuten)
Drücken Sie die folgenden natürlichsprachlichen Aussagen als prädikatenlogische Formeln aus.

1. Es gibt keinen größeren Teiler von x als y.
2. Das Produkt von zwei ungeraden Zahlen ist eine ungerade Zahl.
3. Für jede positive ganze Zahl x gibt es eine positive ganze Zahl y mit $\log(y) > x$.

d) (**, 3 Minuten)
Finden Sie für die folgenden prädikatenlogischen Formeln natürlichsprachliche Aussagen.

1. $\forall x(\forall y((x > 0) \wedge (y > 0) \implies (x \cdot y > 0)))$.
2. $\forall x(\exists z(x \cdot z = 1))$
3. $\forall x((x \mod 2 = 0) \vee (x \mod 2 = 1))$

Aufgabe 43 ** *(Problemspezifikationen, 14 Minuten)*

a) (**, 8 Minuten)
Geben Sie für jede der folgenden natürlichsprachlichen Problemstellungen eine Problemspezifikation an (jeweils 2 Minuten).

1. Berechne den Abstand zweier reeller Zahlen x und y.
2. Bestimme die Fläche eines Rechtecks bei gegebener Breite b und Länge l.
3. Bestimme für zwei gleich lange und nicht leere Wörter u, v über dem Alphabet $\{0, 1\}$, ob u bzgl. der lexikographischen Ordnung kleiner, gleich oder größer als v ist.
4. Berechne, wie oft ein Wert x in einer Folge reeller Zahlen $x_1, \ldots, x_n$ vorkommt.

b) (*, 6 Minuten)
Welches Problem löst jeweils der folgende Algorithmus? Geben Sie eine zugehorige Problemspezifikation an (jeweils 3 Minuten).

1.

```
Eingabe : z_1, ..., z_n, a, f ∈ ℝ
c ← 0;
i ← 1;
solange i ≤ n tue
    wenn |z_i − a| > f dann
        c ← c + 1;
Ausgabe : c
```

2.

```
Eingabe : a_1, ..., a_n, s ∈ ℤ mit ∀i ∈ [1, n − 1](a_i < a_{i+1})
l ← 0;
r ← n + 1;
m ← 0;
solange l ≤ r tue
    m ← l + (r − l)/2;
    wenn a_m = s dann
        Ausgabe : m
    sonst
        wenn a_m < s dann
            l ← m + 1;
        sonst
            r ← m − 1;
Ausgabe : 0
```

Aufgabe 44 *** *(Partielle Korrektheit, 16 Minuten)*

a) (*, Wertzuweisungen, 2 Minuten)

Im Folgenden sei jeweils eine Anweisung und eine Vorbedingung dieser Aweisung gegeben. Formulieren Sie jeweils eine möglichst **restriktive und aussagekräftige** Nachbedingung der Anweisung (jeweils 1 Minute).

1.

$\{a > b\}$
$a \leftarrow a - b;$

2.

$\{i <= n\}$
$i \leftarrow i + 1;$

b) (**, Fallunterscheidungen, 4 Minuten)
Im Folgenden sei jeweils eine Fallunterscheidung und eine Vorbedingung dieser Fallunterscheidung gegeben. Formulieren Sie jeweils eine möglichst **restriktive und aussagekräftige** Nachbedingung mit geeigneten Zwischenzusicherungen nach Eintritt in den **dann**-Block (jeweils 2 Minuten).

1.

$\{c = |\{i \mid i \in [1, k-1], s = a_i\}|\}$
wenn $s = a_k$ **dann**
  $c \leftarrow c + 1;$

2.

$\{\forall i \in [1, k](s > a_i)\}$
wenn $(s > a_{k+1})$ **dann**
  $k \leftarrow k + 1;$

c) (***, Schleifeninvarianten, 4 Minuten)

Im Folgenden sei jeweils eine Schleife und eine Vorbedingung dieser Schleife gegeben. Zeigen Sie jeweils, dass es sich bei der Vorbedingung um eine Schleifeninvariante handelt, indem Sie geeignete Zwischenzusicherungen nach Eintritt in den Schleifen-Block formulieren, und geben Sie eine möglichst **restriktive und aussagekräftige** Nachbedingung der Schleife an (jeweils 2 Minuten).

1.

$\{i \leq n\}$
solange $(i < n)$ **tue**
  $i \leftarrow i + 1;$

2.

$\{p = (i-1)!\}$
solange $(i \leq n)$ **tue**
  $p \leftarrow p \cdot i;$
  $i \leftarrow i + 1;$

d) (***, Partielle Korrektheit, 6 Minuten)

Sei folgende Problemspezifikation gegeben:
Eingabe: $a \in \mathbb{N}, a > 0$
Ausgabe: $z \in \mathbb{N}, z > 0$
Funktionaler Zusammenhang: $z = 6 \cdot a$

Weisen Sie die partielle Korrektheit des nachfolgenden Algorithmus bzgl. dieser Problemspezifikation nach, indem Sie geeignete Zusicherungen einfügen.

Eingabe : $a \in \mathbb{N}^+$
$z \leftarrow 0;$
$i \leftarrow 1;$
solange $(i \leq a)$ **tue**
  $z \leftarrow z + 6;$
  $i \leftarrow i + 1;$
Ausgabe : z