

Allgemeine Hinweise:

- Die **Deadline** zur **Abgabe** der Hausaufgaben ist am **Donnerstag, den 28.10.2021, um 18:00 Uhr**.
- Der **Workflow** sieht wie folgt aus. Die Abgabe der Hausaufgaben erfolgt **im Moodle-Lernraum** und kann nur in **Zweiergruppen** stattfinden. Dabei müssen die Abgabepartner*innen **dasselbe Tutorium** besuchen. Nutzen Sie ggf. das entsprechende **Forum** im Moodle-Lernraum, um eine*n Abgabepartner*in zu finden. Es darf **nur ein*e** Abgabepartner*in die Abgabe hochladen. Diese*r muss sowohl die **Lösung** als auch den **Quellcode** der Programmieraufgaben hochladen. Die Bepunktung wird dann von uns für **beide** Abgabepartner*innen **separat** im Lernraum eingetragen. Die Feedbackdatei ist jedoch nur dort sichtbar, wo die Abgabe hochgeladen wurde und muss innerhalb des Abgabepaars **weitergeleitet** werden.
- Die **Lösung** muss als PDF-Datei hochgeladen werden. Damit die Punkte beiden Abgabepartner*innen zugeordnet werden können, müssen **oben** auf der **ersten Seite** Ihrer Lösung die **Namen**, die **Matrikelnummern** sowie die **Nummer des Tutoriums** von **beiden** Abgabepartner*innen angegeben sein.
- Der **Quellcode** der Programmieraufgaben muss als **.zip-Datei** hochgeladen werden und **zusätzlich** in der PDF-Datei mit Ihrer Lösung enthalten sein, sodass unsere Hiwis ihn mit Feedback versehen können. Auf diesem Blatt müssen Sie noch keine Codeabgabe machen.
- Einige Hausaufgaben müssen im Spiel **Codescape** gelöst werden. Klicken Sie dazu im Lernraum rechts im Block “Codescape” auf den angegebenen Link. Diese Aufgaben werden getrennt von den anderen Hausaufgaben gewertet.

Tutoraufgabe 1 (Überblickswissen):

- Was passiert beim Kompilieren und Ausführen eines Java-Programms?
- Was ist der Unterschied zwischen der Variable `x`, dem `char`-Wert `'x'` und dem `String`-Wert `"x"`?
- Woran erkennt man bei einer im Zweierkomplement dargestellten Zahl, ob diese positiv oder negativ ist?

Tutoraufgabe 2 (Syntax und Semantik):

- Die Menge der syntaktisch korrekten einfachen arithmetischen Ausdrücke (**EAA**) wird durch die Grammatik $G_1 = (\{S_1\}, \{(\cdot, \cdot), \cdot, \text{plus}, s, \mathcal{O}\}, P_1, S_1)$ definiert, wobei P_1 genau die folgenden Produktionsregeln enthält:

$$\begin{aligned} S_1 &\rightarrow \mathcal{O} \\ S_1 &\rightarrow s(S_1) \\ S_1 &\rightarrow \text{plus}(S_1; S_1) \end{aligned}$$

Die Semantik $\mathcal{W}(\mathcal{A})$ eines syntaktisch korrekten **EAA**s \mathcal{A} ist wie folgt definiert, wobei x und y ebenfalls syntaktisch korrekte **EAA**s sind:

$$\begin{aligned} \mathcal{W}(\mathcal{O}) &= 0 \\ \mathcal{W}(s(x)) &= \mathcal{W}(x) + 1 \\ \mathcal{W}(\text{plus}(x; y)) &= \mathcal{W}(x) + \mathcal{W}(y) \end{aligned}$$

Für alle **EAA**s \mathcal{A} gilt also $\mathcal{W}(\mathcal{A}) \in \mathbb{N}$.

Geben Sie für die folgenden drei Ausdrücke an, ob es sich um einen syntaktisch korrekten **EAA** handelt und welche Semantik er hat.

- i) $\text{plus}(s(\mathcal{O}); \mathcal{O})$ ii) $\text{plus}(\mathcal{O}; s(\mathcal{O}); \mathcal{O})$ iii) $\text{plus}(\text{plus}(\mathcal{O}; s(\mathcal{O})); s(\text{plus}(\mathcal{O}; \mathcal{O})))$

- b) Begründen oder widerlegen Sie: Zwei Ausdrücke mit gleicher Syntax haben auch die gleiche Semantik.
c) Begründen oder widerlegen Sie: Ein syntaktisch korrektes Programm ist auch semantisch korrekt.

Aufgabe 3 (Syntax und Semantik):

(6 + 3 + 3 = 12 Punkte)

- a) Die Menge der syntaktisch korrekten **SASP** Programme wird durch die Grammatik $G_2 = (\{A, B, S_2\}, \{., :-, p, q, r, s\}, P_2, S_2)$ definiert, wobei P_2 genau die folgenden Produktionsregeln enthält:

$$\begin{aligned} S_2 &\rightarrow A. \\ S_2 &\rightarrow A. S_2 \\ A &\rightarrow B \\ A &\rightarrow B :- B \\ B &\rightarrow p \\ B &\rightarrow q \\ B &\rightarrow r \\ B &\rightarrow s \end{aligned}$$

Die Semantik $\mathcal{W}(\mathcal{P})$ eines syntaktisch korrekten **SASP** Programms \mathcal{P} ist wie folgt definiert, wobei \mathcal{P}' ebenfalls ein syntaktisch korrektes **SASP** Programm ist und $x, y \in \{p, q, r, s\}$:

$$\begin{aligned} \mathcal{W}(x.) &= \{x\} \\ \mathcal{W}(x :- y.) &= \emptyset \\ \mathcal{W}(\mathcal{P}' x.) &= \mathcal{W}(\mathcal{P}') \cup \{x\} \\ \mathcal{W}(\mathcal{P}' x :- y.) &= \begin{cases} \mathcal{W}(\mathcal{P}') \cup \{x\} & \text{falls } y \in \mathcal{W}(\mathcal{P}') \\ \mathcal{W}(\mathcal{P}') & \text{sonst} \end{cases} \end{aligned}$$

Für alle **SASP** Programme \mathcal{P} gilt also $\mathcal{W}(\mathcal{P}) \subseteq \{p, q, r, s\}$.

Geben Sie für die folgenden Ausdrücke an, ob es sich um ein syntaktisch korrektes **SASP** Programm handelt und welche Semantik es hat.

- i) $t. s :- t.$ ii) $s. p :- r. r. q :- r.$ iii) r iv) $p. q. q :- r. r. s :- q.$

- b) Begründen oder widerlegen Sie: Zwei Ausdrücke einer Sprache mit unterschiedlicher Semantik haben auch immer eine unterschiedliche Syntax.
c) Begründen oder widerlegen Sie: Zwei Ausdrücke einer Sprache mit unterschiedlicher Syntax haben auch immer eine unterschiedliche Semantik.

Tutoraufgabe 4 (Formale Sprachen und Grammatiken):

Gegeben sei die folgende Sprache:

$$L_1 = \{w \in \{a, b\}^* \mid \text{auf ein } a \text{ folgt nie ein } b \text{ oder auf ein } b \text{ folgt nie ein } a\}$$

Die folgenden Wörter sind beispielsweise in der Sprache enthalten:

$$aaab \quad bbaa \quad aa \quad \varepsilon$$

Folgende Wörter sind nicht Bestandteil der Sprache:

bab abba baba

- Geben Sie eine kontextfreie Grammatik an, welche die Sprache L_1 erzeugt.
- Geben Sie eine Grammatik in EBNF an, die L_1 definiert. Ihre Grammatik darf nur aus einer Regel bestehen und diese Regel darf nicht rekursiv sein (d. h. das Nichtterminalsymbol auf der linken Seite darf rechts nicht auftreten).

Um die Lesbarkeit zu erhöhen, dürfen Sie Anführungszeichen um Terminalsymbole weglassen.

- Geben Sie ein Syntaxdiagramm ohne Nichtterminalsymbole an, das die Sprache L_1 definiert.

Aufgabe 5 (Formale Sprachen und Grammatiken): (5 + 2 + 2 = 9 Punkte)

Gegeben sei die folgende Sprache:

$L_2 = \{w \in \{a, b\}^* \mid \text{auf jedes } b \text{ folgen direkt mindestens drei } a \text{ oder auf jedes } a \text{ folgt direkt mindestens ein } b\}$

Die folgenden Wörter sind beispielsweise in der Sprache enthalten:

b abbbbabbb aaaabaaabaaa ε

Folgende Wörter sind nicht Bestandteil der Sprache:

aba bbbbbbba bababa abbaba

- Geben Sie eine kontextfreie Grammatik an, welche die Sprache L_2 erzeugt.
- Geben Sie eine Grammatik in EBNF an, die L_2 definiert. Ihre Grammatik darf nur aus einer Regel bestehen und diese Regel darf nicht rekursiv sein (d. h. das Nichtterminalsymbol auf der linken Seite darf rechts nicht auftreten).

Um die Lesbarkeit zu erhöhen, dürfen Sie Anführungszeichen um Terminalsymbole weglassen.

- Geben Sie ein Syntaxdiagramm ohne Nichtterminalsymbole an, das die Sprache L_2 definiert.

Tutoraufgabe 6 (Zweierkomplement):

- Sei x eine ganze Zahl. Wie unterscheiden sich die Zweierkomplement-Darstellungen von x und $-x$?
- Erklären Sie im Detail, wie die beiden Ausgaben des folgenden Programms berechnet werden.

```
public class Test {
    public static void main(String[] args) {
        int zahl = -2147483648;

        System.out.println(zahl + 1);
        System.out.println(zahl - 1);
    }
}
```

Hinweis: $-2^{31} = -2147483648$

- Welche Zahlen repräsentieren die folgenden Bitfolgen im 5-Bit Zweierkomplement?

00010 10111 11011 01101 10000

Aufgabe 7 (Zweierkomplement):

(5 + 8 = 13 Punkte)

- a) Welche Zahlen repräsentieren die folgenden Bitfolgen im 10-Bit Zweierkomplement?

0101110100 1010000101 1101001011 0111001100 1010010010

- b) Die zwei folgenden Java-Ausdrücke werten jeweils zu `true` aus. Geben Sie dafür jeweils eine Begründung.

1) `2_147_483_000 + 648 == -(-2_147_483_648)`

2) `2_000_000_000 + 1_000_000_000 + 1_294_967_296 == 0`

Hinweise:

- In Java dürfen in einer Zahl beliebig viele `_` zwischen den Ziffern auftreten. Dies verändert den Wert der Zahl nicht, sondern es wird nur verwendet, um die Lesbarkeit der Zahl zu erhöhen. `2_000_000_000` ist also nur eine andere Schreibweise für die Zahl 2000000000.

Tutoraufgabe 8 (Casting):

Bestimmen Sie den Typ und das Ergebnis der folgenden Java-Ausdrücke und begründen Sie Ihre Antwort. Sollte der Ausdruck nicht typkorrekt sein, begründen Sie, worin der Fehler besteht.

Dabei seien die Variablen `x`, `y` und `z` wie folgt deklariert: `int x = 1; int y = 2; int z = 3;`

- a) `false && true`
- b) `10 / 3`
- c) `10 / 3.`
- d) `x == y ? x > y : y < z`
- e) `(byte) (127 + 1)`
- f) `'x' + y + z`
- g) `x + y + "z"`
- h) `1 || 0`

Aufgabe 9 (Casting):

(16 Punkte)

Bestimmen Sie den Typ und das Ergebnis der folgenden Java-Ausdrücke und begründen Sie Ihre Antwort. Sollte der Ausdruck nicht typkorrekt sein, begründen Sie, worin der Fehler besteht.

Dabei seien die Variablen `x`, `y` und `z` wie folgt deklariert: `int x = 1000000000; int y = 121; int z = 126;`

- a) `2000000000 + x`
- b) `2000000000 + 'x'`
- c) `2000000000 + "x"`
- d) `(byte) (3 * z) == 'z' || false`
- e) `(int) 2147483648L * z`
- f) `(byte) 256 * 3f`
- g) `"x" + y - z`
- h) `y != 'y' ? 1.0 : 'z'`

Aufgabe 10 (Intro, Deck 0 und Deck 1):

(Codescape)

Schließen Sie das Intro und das Tutorial zum Spiel Codescape ab und lösen Sie die Missionen von Deck 0 und Deck 1.

Ihre Lösung für die Codescape Missionen wird nur dann für die Zulassung gezählt, wenn sie Ihre Lösung vor der einheitlichen Codescape Deadline am Samstag, den 22.01.2022, um 23:59 Uhr abschicken.