Theoretical Computer Science

Winter semester 2024/25

Prof. Dr. Georg Schied

Solution - Assignment 11

Deadline: Monday, 20 January 2025

10 out of 20 points have to be achieved in order to pass

Last assignment with obligabory exercises.
Reminder: The following achievements are required as **exam prerequisite**:

- **10 assignments** must be passed **and**
- **20 quizzes** must be successfully solved.

Exercise 11.1

The following grammar defines arithmetic expressions, consisting of numbers, operators + and *, and brackets. As usual, operation * has higher precedence than +, and both operators are handled as being left-associative.

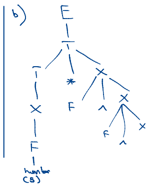
$$E \rightarrow \quad E + T \mid T$$
$$T \rightarrow \quad T * F \mid F$$
$$F \rightarrow \quad (E) \mid \textbf{number}$$

Follow precedence

a) $E \rightarrow E + T$
$\mid T$
$T \rightarrow T * X$
$X \rightarrow F \wedge X$
$\mid F$
$F \rightarrow (E)$
$\mid$ number

E +
T *
X ^
F

b)



a) Augment the grammar with a power operator ^, where `3^5` denotes 3^5. The power operator ^ should have higher precedence than multiplication * and it should be right-associative, e.g. `5*4^3^2` should be read as `5 * (4 ^ (3 ^ 2))`.

b) Draw the derivation tree for `5*4^3^2` according to the extended grammar (2, 3, 4, 5 are considered number symbols).

Exercise 11.2

a) The following grammar is not LL(1). Give three reasons why not, without computing the predictive parsing table.

Left recursion

$$S \rightarrow \quad AaS \mid AB$$
$$A \rightarrow \quad A{+}A \mid a$$
$$B \rightarrow \quad Bb \mid cc$$

Left Recursion

Right recursion is not a problem in top down approach

b) Transform the following grammar so that it is no longer left recursive.

$$S \rightarrow \quad bSb$$
$$\mid \quad AcA$$
$$A \rightarrow \quad bb$$
$$\mid \quad AbA$$
$$\mid \quad AaS$$

Step (I) EBNF

$A \rightarrow bb(aS \mid SA)^*$

Step (II) context free

$A \rightarrow bbR$

$R \rightarrow \varepsilon \mid aSR \mid bAR$

Exercise 11.3 - obligatory (5 points)

a) The following grammar does not have the LL(1) property. Indicate two reasons why not (without computing the predictive parsing table).

$$
\begin{aligned}
S \rightarrow\ & SAa \\
|\ & Bb \\
|\ & Sc \\
A \rightarrow\ & aa \\
B \rightarrow\ & bAcc \\
|\ & bAa \\
|\ & ca
\end{aligned}
$$

b) Transform the grammar so that it is LL(1).

Exercise 11.4

a) Which of the following propositional formulas are *satisfiable*, which are *valid* (*tautologies*), which are *unsatisfiable*?

(1) (square → ¬ heavy) ∨ square   valid / tautology

(2) red ∧ square → ¬ heavy   statisfiable

(3) ¬ square ∧ ¬ heavy ∧ (heavy ∨ square)   unsatisfible

p	q	p→q
T	T	T
T	F	F
F	T	T
F	F	T

b) Are both propositional logic formulas equivalent? (give a short justification)

(1) ¬ (p ∧ (q ∨ r)) and ¬p ∨ ¬q ∧ ¬r   → ¬p∨¬(q∨r) ⇒ ¬p∨¬q∧¬r

(2) a → b and ¬a → ¬b   ¬¬b∨¬a → b∨¬a

F T →F   F ¬T F →T

(3) a → b and ¬b → ¬a   → ¬¬a∨b ⟷

c) Simplify the following propositional formulas as far as possible by equivalent transformations.

(b → a) ∨ (b → ¬a)   → a ∨ ¬a ∨ ¬b ∨ ¬b

(¬b ∨ a) ∨ (¬b ∨ a)   true ∨ ¬b

True

Exercise 11.5 - obligatory (4 points)

a) Is the following propositional formula valid, satisfiable or unsatisfiable?

(heavy ∧ ¬ red) ∨ (¬ heavy ∧ red)

b) Simplify the following propositional formula as far as possible by equivalent transformations. Indicate the individual steps.

Exercise 11.6

Let the following vocabulary for first-order logic formulas be given:

- Function symbols: `f` (unary), `g` (binary)
- Predicate symbols: `P` (unary), `Q` (binary)
- Constant symbols: a, b
- Variable symbols: x, y, z

Exercise 11.3 - obligatory (5 points)

a) The following grammar does not have the LL(1) property. Indicate two reasons why not (without computing the predictive parsing table).

S → SAa
  | Bb
  | Sc

A → aa

B → bAcc
  | bAa
  | ca

b) Transform the grammar so that it is LL(1).

①

$$\left.\begin{array}{l} B \rightarrow \underline{bA}\,cc \\ \quad\;\; \underline{bA}\,a \end{array}\right\}$$ Ambiguous, there is a conflict for non-terminal B and bA

②

$$\left.\begin{array}{l} S \rightarrow SAa \\ \quad\;\; Sc \end{array}\right\}$$ S has left recursion

b) Fixing common initial parts

$B \rightarrow bAR$

$R \rightarrow cc \mid a$

② $S \rightarrow (Bb)(SAa \mid Sc)^*$

$S \rightarrow Bb$
$\quad\;\; \mid R$

$R \rightarrow$

Exercise 11.5 - obligatory (4 points)

a) Is the following propositional formula valid, satisfiable or unsatisfiable?

(heavy ∧ ¬ red) ∨ (¬ heavy ∧ red)

b) Simplify the following propositional formula as far as possible by equivalent transformations. Indicate the individual steps.

a) its satisfiable because, some of the results in the preposition table has a combinations of True and False

heavy	red	
0	0	→ 1
0	1	1
1	0	→ 0
1	1	

b) $$\neg(\neg(heavy \wedge \neg red) \wedge \neg(\neg heavy \wedge red))$$

$$\neg(\neg heavy \vee red \wedge heavy \vee \neg red)$$

$$\neg(heavy \rightarrow red \wedge \neg(\neg heavy \wedge red))$$

$$\neg(heavy \rightarrow red \wedge \neg(heavy \rightarrow red))$$

$$(\neg heavy \rightarrow \neg red) \vee (heavy \rightarrow red)$$

$$\neg(heavy \rightarrow red) \vee (heavy \rightarrow red)$$

$\forall z \exists y$ is valid

$\exists x \exists y$ is valid

Are the following first-order formulas **syntactically correct** or not?

(1) `Q(f(x), g(a))` binary X

(2) `∀z Q(b, f(z)) → ∃y Q(P(b),P(y))` Predicate can't be in a predicate

(3) `∃y ¬( ∀z P(f(x)) ∧ ∀y Q(a,y) )` correct

(4) `∀y Q( f(y) ∨ g(a,b) )` Cant have operator with functions like

Exercise 11.7 - obligatory (5 points)

Let the same vocabulary for first-order logic formulas be given as in the previous exercise. Are the following first-order formulas **syntactically correct** or not? Indicate all errors.

(1) `∃y P(f(x,b)) → ∀z Q(g(z,b))`

(2) `∀x∃z Q(g(x,b), z) ∨ ∀z P(f(z))`

(3) `¬∃∀x Q(b, g(a,x))`

(4) `∀x Q(x, f(z)) → ∃z f(z)`

(5) `∀y Q(y, ∃x P(g(a,x)))`

Exercise 11.8

Let the following predicates on persons be given:

Male(p)	p is male
Brother(p_1, p_2)	p_1 is brother of p_2
Child(c, p)	c is a child of p (i.e. p is parent of c)
Aunt(a, p)	a is aunt of p

a) Explain in everyday language the meaning of the following formulas. Are the formulas true if they are considered as statements about all living persons?

(1) $\neg\forall p$ Male(p)

(2) $\forall p\ \exists b$ Brother(b, p)

(3) $\exists p\ \exists b$ Brother(b, p)

(4) $\exists p\ \forall b$ Brother(b, p)

(5) $\forall p\ \forall b$ Brother(b, p) . 1 = Every persons is Everyones brother

b) Formulate as a first-order formula:

(1) "There is someone who is male." $\exists p$ Male(p)

(2) If a person p_1 is brother of p_2, then also p_2 is brother of p_1. $\forall p_1 \forall p_2$ Brother(p_1,p_2) → Brother(p_2,p_1)

(3) "Not every person has a child." $\neg \forall p \exists c$(Child(c,p))

(4) "A male person is never aunt of someone else." $\forall p$ (Male(p)) → $\exists q$ Aunt(p,q)

(5) "If any two persons are brothers, then they have a common parent."

(6) "If a person is brother of someone else, then the person is male."

$\forall p_1 \forall p_2$ (Brother(p_1, p_2) → $\exists p$(Child(p_1,q) ∧ Child(p_2,q))

6) $\forall p$($\exists q$ Brother(p,q) → Male(p)) somebody

Ex 11.7

Let the following vocabulary for first-order logic formulas be given:

- Function symbols: f (unary), g (binary)
- Predicate symbols: P (unary), Q (binary)
- Constant symbols: a, b
- Variable symbols: x, y, z

Exercise 11.7 - obligatory (5 points)

Let the same vocabulary for first-order logic formulas be given as in the previous exercise. Are the following first-order formulas **syntactically correct** or not? Indicate all errors.

(1) $\exists y\ P(f(x,b)) \rightarrow \forall z\ Q(g(z,b))$

(2) $\forall x \exists z\ Q(g(x,b), z) \vee \forall z\ P(f(z))$

(3) $\neg \exists \forall x\ Q(b, g(a,x))$

(4) $\forall x\ Q(x, f(z)) \rightarrow \exists z\ f(z)$

(5) $\forall y\ Q(y, \exists x\ P(g(a,x)))$

1) not correct, Q needs to accept binary parameters

2) correct

3) not correct, variable for existential quantifier is missing

4) correct

5) not correct

Exercise 11.9

Let the following vocabulary for first-order formulas be given. We assume natural numbers ℕ as domain of discourse.

Constants symbols:	0, 1, 2	
Variable symbols:	n, m, k	
Function symbols:	+	(addition, binary),
	·	(multiplication, binary)
Predicate symbols:	=	(equality, binary),
	<	(less-than relation, binary),
	isSquare	(is a square number, unary)

As usual, +, ·, =, and < can be written in infix notation.

a) There is a number *n*, so that 2*n*+*1* is a square number. $\exists n\ Square(2 \cdot n + 1)$

b) For each square number there also exists a larger square number. $\forall n\ (square(n) \rightarrow \exists m\ n < m \wedge Square(m))$

c) The product of two square numbers is always also a square number. $\forall n \forall m\ (Square(n) \wedge Square(m) \rightarrow Square(n \cdot m))$

d) There exists no number that is greater than every square number. $\exists n\ \forall m\ (Square(m) \rightarrow m < n))$

Exercise 11.10 - obligatory (6 points)

Let the same vocabulary and same domain of discourse ℕ for first-order formulas be given as in Exercise 11.6. Formulate the following statements about natural numbers as first order formulas:

a) There exists at least one number that is not square.

b) For each number *n* is *n*·*n* a square number.

c) Not every number is a square number.

d) For each *n* that is a square number, *n*+1 is not a square number.

e) For each number there is a greater square number.