Duale Hochschule Baden-Württemberg Stuttgart

Klausur: Theoretische Informatik I

(Lehrveranstaltung)

Kurs: …TINF18AIA……………………….

Dozent: Prof. Dr. Bernd Schwinn

Studierende(r)
(Matr.Nr.): ....

Semester: 1.

Hilfsmittel: keine **Dauer: 100 Min.**

Bewertung: Erreichte Punktzahl: Maximale Punktzahl: 85

Note: .... Signum: ....

Anmerkungen:

Aufgabennr.:		maximale Punkte	erreichte Punkte	Bemerkungen
1		6		
2		5		
3		6		
4		9		
5		8		
6		8		
7		6		
8		8		
9		8		
10		6		
11		9		
12		6		
Summe		**85**		

Task 1: (6 Points)

Express the following phrases by logical formulae:

- The successor of each natural number is either even or odd
- The root of a natural square number is a natural number
- A natural number is a power of two if there exists a natural number that defines the number of times to multiply 2 with itself to get this number
- The sum of two subsequent natural square numbers is odd

Task 2: (5 Points)

Presume a signature with the sort symbols ‘s1’ and ‘s2’,
a 2-ary function f(s2,s1) -> s1
a 3-ary function g(s1,s2,s1) -> s2
a 2-ary predicate P(s1,s2)
a 3-ary predicate Q(s2,s1,s2)
, variables x, y and z of sort s1 as well as a, b and c of sort s2.
Which of the following expressions are formulae in predicate logic? If not, give the reason why!

a) $\neg Q(g(z, b, f(g(x, g(y, a, x), y), f(g(y, c, x), z))), z, g(z, a, z)) \vee P(g(f(g(y, c, z), x), c, x), f(g(z, a, y), x))$

b) $Q(g(y, g(z, a, x), z), f(g(y, b, x), f(a, x))) \wedge P(f(g(f(b, f(c, z)), b, x), y), g(f(g(z, b, y), f(g(x, a, y), z)), b, y))$

c) $Q(a, f(g(x, a, f(a, x)), f(c, f(g(x, c, z), f(g(f(g(x, a, y), x), a, f(b, g(y, b, z))), x)))), b) \rightarrow P(z, g(x, g(y, c, z),x))$

d) $Q(a, f(g(z, g(f(c, x), g(f(b, y), b, f(g(x, b, y), f(a, f(g(z, b, y), x)))), f(g(x, a, z), z)), f(g(f(c, z), a, x), y)), z), b)$

e) $Q(g(z, c, g(x, b, y)), f(g(y, b, x), z), g(y, b, f(b, z))) \leftrightarrow P(f(g(f(a, z), b, x), y), g(z, g(z, a, z), y))$

Task 3: (6 Points)

Presume the following sets of formulae:

$X = \{P \wedge S \rightarrow Q, \neg R \leftrightarrow Q, P \wedge \neg(S \vee \neg R)\}$

$Y = \{\neg R \wedge S \rightarrow \neg P, R \vee (\neg Q \vee S), R \rightarrow \neg P \wedge Q\}$

$Z = \{(Q \wedge \neg S) \vee P, (R \vee \neg S) \rightarrow (\neg P \wedge R), P \vee S \rightarrow \neg Q\}$

For each set determine all the replacements that makes it true. State all possible logical conclusions among the sets of formulae!

Task 4: (3 + 3 + 3 Points)

Transform the following formulae into conjunctive normal form and notate **all** resulting Gentzen-formulae.

a) $(R \leftrightarrow \neg S) \rightarrow Q \vee P$

b) $(S \wedge \neg P) \leftrightarrow \neg (R \vee \neg Q)$

c) $(S \wedge \neg Q \rightarrow R \wedge \neg P) \rightarrow Q$

Task 5: (3 + 5 Points)

Presume the following formulae. First transform them into Prenex normal form and scolemise them finally.

a) $\exists y\, (P(x, y) \vee \exists x\, \neg R(x, y, x) \rightarrow \neg\forall z\, Q(x, y, z)) \wedge \exists z\, \forall y\, R(x, y, z)$

b) $\neg\exists x\, (\exists z\, P(x, y, z) \leftrightarrow \forall y\, Q(x, y, z)) \wedge \neg\forall x\, R(z, y, x)$

Task 6: (8 Points)

Prove the correctness of the following specification

```
{ X>Y ; Y>=0; X, Y of type integer }
        I := X;
        Z:=0;
        repeat
                Z := Z + 2;
                I := I - 1
        until I = Y
{ Z=2*(X-Y) ; X>Y ; Y>=0 ; X, Y of type integer }
```

Task 7: (6 Points)

Presume the following statements with postcondition

```
if X<Y then X := Y - X;
if X<0 then Y := X - Y
       else  Y := X + Y
```

{Y>X; X, Y of type integer }

Determine the weakest precondition.

Task 8: (8 Points)

Are the following pairs of literals unifiable? If yes, give the bindings resulting from unification. If not, give the reason why.

a) p1([[X|Y]|[X,Y]],[[Z]|X]) p1([A|[B|B]],C)

b) p2([[X|R]|Y],[Y|X]) p2([Y|A],[Y])

c) p3([X|[[Y]|[]]],[[Y]|X]) p3([A|R],[R|A])

d) p4([X,[Y]|[[X]]],[X]) p4([A|[B,[A]]],A)

e) p5(X,[Y|[Y|X]]) p5([A],[B|[A|B]])

Task 9: (5 + 3 Points)

Define a Prolog program that is analogous to the Lisp function mystery given below.

```
(defun mystery (L)
    (cond ((null L) L)
          ((atom L) nil)
          ((null (cdr L)) nil)
          ((> (car L) (cadr L)) (cons (car L) (cons (cadr L) (mystery (cddr L)))))
          (T (mystery (cddr L))))
```

b) Determine for the Lisp-Funktion given in a) the number of executed test operations, if the function is activated with a list containing n (n>0, n is even) numbers. Start the cost calculation with a recursion equation and solve it finally.

Task 10: (6 Points)

Define Prolog clauses for a predicate oddpossum that sums up the values of the numbers at the odd positions within a list of integer-numbers.

Task 11: (2 + 3 + 1 + 3 Points)

Presume L to be a list containing at least two (non-empty) sublists of (at least two) numbers. Give a Lisp expression that

a) exchanges the first two sublists if the first number of the second sublist is negative

b) calculates the sum of the first number of the first and the first number of the second sublist and inserts this into a new sublist located in front of the remaining sublists

c) sorts the first as well as the second sublist and joins them zipper-like in a newly created first sublist (followed by the 3rd, 4th, … sublist)

d) calculates the sum of the last number of the first sublist and the last number of the second sublist and inserts this sum to the beginning of the second sublist

Task 12: (6 Points)

Define a Lisp function oddpossum (analogous to Task 10) that sums up the values of the numbers at the odd positions within a list of integer numbers.

Example: (oddpossum ‘(1 -8 0 3 -3)) returns -2