Task 1: (6 Points)

Express the following phrases by logical formulae:

- The half of an even natural number is a natural number as well
- An odd natural number has an even predecessor and an even successor
- The successor of a natural number is odd or even
- A square number is a natural number that is the product of a natural number with itself

Task 2: (5 Points)

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

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

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

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

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

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

Task 3: (6 Points)

Presume the following sets of formulae:

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

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

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

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

Task 4: (3 + 3 + 4 Points)

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

a) $R \leftrightarrow S \vee \neg Q \wedge P$

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

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

Task 5: (3 + 5 Points)

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

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

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

Task 6: (8 Points)

Prove the correctness of the following specification

```
{ X>Y ; Y>=0; X, Y of type integer }
        I := 0;
        Z:=X;
        while I<Y do
        begin  I := I + 1;
               Z := Z + 1
        end
{ Z=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)

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

```
(defun  mystery (L)
        (cond  ((listp L)     (cond  ((null L)         0)
                                     ((< (car L) 0)  (- (mystery (cdr L)) 1))
                                     (T                (+ (mystery (cdr L)) 1)))
               (T      nil)))
```

b) Determine for the Lisp function given in a) the number of executed test functions if the function is activated with n (n>0) positive numbers. Start the cost calculation with a recursion equation and solve it finally.

Task 10: (6 Points)

Define Prolog clauses for a predicate modify that deletes all ’0’s within a list of numbers and that replaces each negative number by -1.

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

Assume L to be a list containing at least two (non-empty) sublists (with 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 behind the first two sublists

c) sorts the first as well as the second sublist and joins them zipperwise in a newly created first sublist (followed by the 3^{rd}, 4^{th}, … 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 modify (analogous to Task 10), that deletes all ’0’s within a list of numbers and replaces each negative number by -1.

Example: (modify ‘(1 -8 0 -3 3)) returns (1 -1 -1 3)

Task 1: (6 Points)

Express the following phrases by logical formulae:

- The successor of each natural square number is a natural number
- The result of dividing an even natural number by 2 is a natural number
- Each natural square number is not negative
- The sum of the predecessor and the successor of a prime number is even

Task 2: (5 Points)

Presume a signature with the sort symbols ‘s1’ and ‘s2’,
a 2-ary function f(s1,s2) -> s2
a 3-ary function g(s2,s1,s2) -> s1
a 2-ary predicate P(s2,s1)
a 3-ary predicate Q(s1,s2,s1)
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) $Q(x, f(g(f(y, a), z, b), f(g(b, g(c, x, f(y, a))), a)), y) \lor \neg P(f(g(c, z, f(g(f(z, c), y, b), a)), b), g(a, g(b, y, b), c))$

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

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

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

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

Task 3: (6 Points)

Presume the following sets of formulae:

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

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

Task 4: (3 + 3 + 4 Points)

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

a) $R \lor \neg(S \rightarrow \neg Q \land P)$

b) $\neg(S \lor (P \leftrightarrow R) \land \neg Q)$

c) $S \lor \neg Q \rightarrow (R \leftrightarrow \neg P) \lor Q$

Task 5: (3 + 5 Points)

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

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

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

Task 6: (8 Points)

Prove the correctness of the following specification

```
{ X>Y ; Y>=0; X, Y of type integer }
        I := Y;
        Z:=0;
        repeat
                I := I + 1;
                Z := Z + 1
        until I = X
{ Z=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 begin if** $X<0$ **then** $Y := X - Y$ **end**
>   **else** $Y := Y - X$;
> $Y := X - Y$;

{ $Y>X$; 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]],[Y|Y]) p1([A|B],A)

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

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

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

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

Task 9: (5 + 3 Points)

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

```
mystery([],0).
mystery([X],X).
mystery([A,B|R],N) :- mystery(R,M), N is M+A.
```

b) Determine for the Prolog predicate given in a) the number of **all** executed unifications of literals if the predicate is activated with a list containing n positive numbers whereby n is odd . Start the cost calculation with a recursion equation and solve it finally.

Task 10: (6 Points)

Define Prolog clauses for a predicate possum that sums up all positive numbers out of a list of integer-numbers.

Task 11: (2 + 3 + 1 + 2 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) replaces within the first sublist the first two numbers by the sum of these two numbers

b) calculates the sum of the first number of the first sublist and the second number of the second sublist and inserts the result to the beginning of the second sublist

c) deletes the first sublist and inserts the reverse of this first sublist to the end of L

d) determines the length of the second sublist and inserts this to the end of the first sublist

Task 12: (6 Points)

Define a Lisp function possum (analogous to Task 10) that calculates for a list of integer numbers the sum of all positive numbers contained in it.

Example: (possum ‘(1 -8 0 -3 3)) returns 4

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 3^{rd}, 4^{th}, … 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

Task 1: (6 Points)

Express the following phrases by logical formulae:

- The predecessor of an odd natural number is an even natural number
- The root of a natural square number is a natural number or it is negative
- The sum of the negative root and the positive root of a natural square number is zero
- The sum of the predecessor and the successor of a natural square numbers is even

Task 2: (5 Points)

Presume a signature with the sort symbols 's1' and 's2',
a 3-ary function f(s1,s2,s1) -> s2
a 3-ary function g(s2,s1) -> s1
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) $Q(b, f(z, f(g(a, g(a, x)), b, g(b, g(c, x))), g(a, z))) \lor P(g(b, f(g(c, z), a, y)), f(g(a, y), b, z))$

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

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

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

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

Task 3: (6 Points)

Presume the following sets of formulae:

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

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

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

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 + 4 Points)

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

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

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

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

Task 5: (3 + 5 Points)

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

a) $\forall x\,(\neg P(x,y) \lor \exists y R(x,y,y)) \;\rightarrow \neg(\exists z\; \forall x\; Q(x,y,z) \land \forall y\; R(x,y,x))$

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

Task 6: (8 Points)

Prove the correctness of the following specification

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

Task 7: (6 Points)

Presume the following statements with postcondition

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

{ Y>0; 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|Y]],[]) p1([A,A],A)

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

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

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

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

Task 9: (5 + 3 Points)

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

```
(defun  mystery (L)
        (cond  ((listp L)       (cond  ((null L)         0)
                                       ((< (car L) 0)  (- (mystery (cdr L)) (car L)))
                                       ((> (car L) 0)  (+ (mystery (cdr L)) (car L)))
                                       (T (mystery (cdr L))))
               (T       nil)))
```

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) positive numbers. Start the cost calculation with a recursion equation and solve it finally.

Task 10: (6 Points)

Define Prolog clauses for a predicate **onlypos** that deletes within a list of integer numbers all 0`s and replaces each negative number by its doubled absolute value.

Task 11: (1 + 2 + 2 + 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) joins zipperwise the first two sublists into the first sublist and inserts the result to the end of L

b) reverses and exchanges the first two sublists if the list contains more than two sublists

c) inserts the length of the two first sublists to the end of the according sublist

d) inserts the sum of the first element of the first sublist and the first element of the second sublist to the beginning of the third sublist (if no third sublist exists then a new 3rd sublist with the sum is generated)

Task 12: (6 Points)

Define a Lisp function **onlypos** (analogous to Task 10) that deletes within a list of integer numbers all 0`s and replaces each negative number by its doubled absolute value

Example: (onlypos ‘(1 -8 0 3 0 -3)) returns (1 16 3 6)