

Personnummer: 123456-7890

Namn: p. Carina

Avslutningsprov

Personnummer: 123456-7890

Namn: p. Carina

Avslutningsprov

Fråga 1

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Prolog

Prolog 1.4 poäng

Hur ser den prolog-klausul ut som motsvarar följande satslogiska formel: $a \vee b \rightarrow c$.☐ a, b :- c☐ a or b :- c☐a :- c.
b :- c.☐ c :- a, b.☐ Inget av alternativen är rätt.

Fråga 2

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Prolog

Prolog 2.4 poäng

Vad blir den substitutionsmängd som fås vid följande unificering:
 $\text{zot}(X, Y, f(Y)) = \text{zot}(f(Z), Z, X)$.☐ $\{ X \rightarrow f(f(2)), Y \rightarrow Z, Z \rightarrow f(2) \}$ ☐ $\{ X \rightarrow f(2), Y \rightarrow Z, Z \rightarrow 2 \}$ ☐ $\{ X \rightarrow 2, Y \rightarrow f(2), Z \rightarrow f(2) \}$ ☐ $\{ X \rightarrow f(f(2)), Y \rightarrow 2, Z \rightarrow f(2) \}$ ☐ Inget av alternativen är rätt.

Personnummer: 123456-7890

Namn: p, Carina

AVSLUTAT PROVE

Personnummer: 123456-7890

Namn: p, Carina

AVSLUTAT PROVE

Fråga 3

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Prolog

Prolog 3. 4 poäng

Betrakta följande program:

```
foo([], []).  
foo([1|R], X) :- foo(R, X).  
foo([2|R], [2,2|X]) :- foo(R, X).  
foo([1|R], [1|X]) :- foo(R, X).
```

Vad blir det första resultatet om man har målet `foo([1,2,1,4,2,1,5], X)`?☐ X = [1,2,1,4,2,1,5]☐ X = [1,2,1,2,1]☐ X = [2,2,4,2,2,5]☐ X = [4,5]☐ Inget av alternativen är rätt.

Fråga 4

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Prolog

Prolog 4. 2 poäng

Betrakta följande program:

```
foo([], []).  
foo([1|R], X) :- foo(R, X).  
foo([2|R], [2,2|X]) :- foo(R, X).  
foo([1|R], [1|X]) :- foo(R, X).
```

Vad blir det andra resultatet, dvs om vi inte är nöjda med det första utan backtraccar för att generera ytterligare ett svar, om man har målet `foo([1,2,1,4,2,1,5], X)`?☐ X = [1,2,1,4,2,5]☐ X = [1,2,1,2]☐ X = [2,2,4,2,2,1,5]☐ X = [4,5]☐ Inget av alternativen är rätt.

Personnummer: 123456-7890

Namn: p, Carina

Avsluta provet

Personnummer: 123456-7890

Namn: p, Carina

Avsluta provet

Fråga 5

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Prolog

Prolog 5.2 poäng

Betrakta följande program:

```
bar([], Y, Y).  
bar([H|T], Y, [H|Z]) :- bar(T, Y, Z).
```

Hur många lösningar finns det till målet `bar(X, Y, [1,2,3])`?☐ 1☐ 2☐ 3☐ 4☐ målet saknar lösningar

Fråga 6

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Prolog

Prolog 6.2 poäng

Betrakta följande program:

```
a(1, Y, Z) :- b(Y, 1, c(Z)).  
a(2, 3, 4).
```

☐ b(1).☐ b(2).☐ b(3).☐ c(1).☐ c(2).Hur många lösningar finns till målet `a(X, Y, Z)`?☐ 1☐ 2☐ 3☐ 7☐ målet saknar lösningar

Personnummer: 123456-7890

Namn: p, Carina

Avsluta provet

Personnummer: 123456-7899

Namn: i, Carina

Avsluta provet

Fråga 7

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Prolog

Prolog 7. 2 poäng

Betrakta följande program:

```
lookup([e(N,X)|_], N, X):- !.  
lookup([_|T], N, X):- lookup(T, N, X).
```

Vilka lösningar finns till målet `lookup([e(17,foo), e(42,bar)|H], 42, X)`?

- ☐ X = bar
- ☐ X = bar och H = [e(42,X)|_]
- ☐ oändligt många lösningar
- ☐ målet saknar lösningar

Fråga 8

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Haskell

Haskell 1. 4 poäng

OBS! Denna kod används i Haskellfrågorna 1) - 3).

Betrakta följande kod:

```
myId x = x
```

```
funktion n f  
  | n > 0 = f . funktion (n-1) f  
  | otherwise = myId
```

Vad är en korrekt typsignatur för funktion?

- ☐ funktion :: (Num a, Ord a) => a -> (b -> b) -> b
- ☐ funktion :: (Num a, Ord a) => a -> (a -> a) -> a -> a
- ☐ funktion :: (Num a, Ord b) => a -> (b -> a) -> a
- ☐ funktion :: (Num a, Ord a) => a -> (b -> b) -> b -> b
- ☐ Inget av alternativen är rätt!

Personnummer: 123456-7899

Namn: I, Carina

Avsluta provet

Fråga 9

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Haskell

Haskell 2. 4 poäng

OBS! Denna kod används i Haskellfrågorna 1) - 3).

Betrakta följande kod:

 $\text{myId } x = x$ funktion $n \ f$
$$\begin{cases} n > 0 = f \cdot \text{funktion } (n-1) \ f \\ \text{otherwise} = \text{myId} \end{cases}$$

Vilket/vilka påstående/-n är sanna/sant?

- ☐ anropet funktion 2 $(*)2$ ger en funktion som resultat
- ☐ anropet funktion $(-2) \ (*)2$ 1 ger resultatet 1
- ☐ anropet funktion 2 $(*)2$ ger ett felmeddelande
- ☐ anropet funktion $(-2) \ (*)2$ 1 ger resultatet 4
- ☐ Inget påstående är sant!

Personnummer: 123456-7899

Namn: I, Carina

Avsluta provet

Fråga 10

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Haskell

Haskell 3. 2 poäng

OBS! Denna kod används i Haskellfrågorna 1) - 3).

Betrakta följande kod:

 $\text{myId } x = x$ funktion $n \ f$
$$\begin{cases} n > 0 = f \cdot \text{funktion } (n-1) \ f \\ \text{otherwise} = \text{myId} \end{cases}$$

Vilket/vilka påstående/-n är sanna/sant?

- ☐ funktion använder listomfattning (list comprehensions)
- ☐ funktion är en högre ordningens funktion
- ☐ funktion kan inte ta en anonym funktion som indata
- ☐ funktion är rekursiv men inte svansrekursiv
- ☐ Inget påstående är sant!

Personnummer: 123456-7890

Namn: p, Carina

AVSLUTAT PROVE

Fråga 11

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Haskell

Haskell 4. 4 poäng

Funktionen sorterad är skriven för att kolla om en lista är sorterad i stigande ordning eller ej. Exempelvis ska det gälla att sorterad [1,3,2] ger resultatet False och sorterad [1,2,3] ger resultatet True.

Tyvärr har kodraderna hamnat i oordning och dessutom blandats med rader som är felaktiga (de skrevs under utvecklingsfasen).
Din uppgift är att titta på de föreslagna kombinationerna av rader och avgöra om det resulterar i en korrekt funktion sorterad.

```
Följande rader finns:  
sorterad [] = True -- rad 1  
sorterad [x] = True -- rad 2  
sorterad (x:y) -- rad 3  
sorterad (x:y:resten) -- rad 4  
| x==y = sorterad y -- rad 5  
| x==y = sorterad resten -- rad 6  
| x==y = y : (sorterad resten) -- rad 7  
| x==y = sorterad (y:resten) -- rad 8  
| otherwise = False -- rad 9
```

Vilket/vilka kombinationerna av rader resulterar i en korrekt funktion sorterad?

☐ 1,3,5,9☐ 1,3,8,9☐ 1,2,3,5,9☐ 1,2,3,8,9☐ 1,2,4,6,9☐ 1,2,4,7,9☐ 1,2,4,8,9☐ Inget av alternativen ger en korrekt funktion sorterad!

Personnummer: 123456-7890

Namn: p, Carina

Avsluta provet

Fråga 12

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Haskell

Haskell 5, 2 poäng

OBS! Denna kod används i Haskellfrågorna 5) - 6).

```
Betrakta följande kod:  
import Data.Char  
class Positivt a where  
  myAbs :: a -> a  
instance Positivt Integer where  
  myAbs tal  
    | tal < 0 = (- tal)  
    | otherwise = tal
```

Varför blir resultatet av anropet myAbs (-3.2) ett felmeddelande?

- ☐ man kan inte göra jämförelsen tal < 0 utan derivering (Eq)
- ☐ indata (-3.2) är ej av korrekt typ
- ☐ inget objekt av klassen har skapats
- ☐ myAbs är lokal i klassen Positivt och inte åtkomlig
- ☐ Inget av alternativen är korrekt!

Personnummer: 123456-7899

Namn: L. Carina

Avsluta provet

Fråga 13

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Haskell

Haskell 6. 4 poäng

OBS! Denna kod används i Haskellfrågorna 5) - 6).

Betrakta följande kod:

```
import Data.Char
class Positivt a where
  myAbs :: a -> a
instance Positivt Integer where
  myAbs tal
    | tal < 0 = (- tal)
    | otherwise = tal
```

Man vill nu kunna applicera myAbs på varje element i en lista av längden 2.
Vilket/vilka förslag nedan ger att anropet myAbs [-2,-3] resulterar i [2,3]?

☐

instance Positivt a =>

Positivt [a] where

myAbs lista = map myAbs lista[0:1]

☐

instance Positivt a =>

Positivt (a, a) where

myAbs tupel = (myAbs(fst tupel), myAbs(snd tupel))

☐

instance Positivt a =>

Positivt [a] where

myAbs lista = [myAbs(head lista), myAbs (head (tail lista))]

☐

instance Positivt a =>

Positivt [a] where

myAbs lista = [myAbs(head lista), myAbs (tail lista)]

☐

Inget av alternativen är korrekt!

Personnummer: 123456-7890

Namn: P. Carina

Avsluta provet

Personnummer: 123456-7890

Namn: P. Carina

Avsluta provet

Fråga 14

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Syntax

Syntax 1.3 poäng

Vilken är ordningen i parsningsprocessen?

- ☐ syntaxträd-tecken-slutsymboler-lexikal analys-syntaktisk analys
- ☐ tecken-lexikal analys-slutsymboler-syntaktisk analys-syntaxträd
- ☐ tecken-lexikal analys-syntaxträd-syntaktisk analys-slutsymboler
- ☐ syntaktisk analys-syntaxträd-tecken-lexikal analys-slutsymboler
- ☐ Inget av alternativen är rätt!

Fråga 15

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Syntax

Syntax 2.3 poäng

Vad menas med att DFA och NFA är lika kraftfulla?

- ☐ Varje språk som kan kännas igen av en DFA kan också kännas igen av en NFA och vice versa.
- ☐ En NFA och DFA som känner igen samma språk är lika stora.
- ☐ Tiden att konstruera en DFA och en NFA är lika.
- ☐ Det finns inget språk som kan kännas igen av en NFA men inte av en DFA och vice versa.
- ☐ Inget av alternativen är rätt!

Personnummer: 123456-7890

Namn: P. Carina

Avsluta Döve

Personnummer: 123456-7899

Namn: I. Carina

Avsluta prova

Fråga 16

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Syntax

Syntax 3. 3 poäng

Vilka av nedanstående strängar kan genereras av $x(xy|yx)^*y$

☐ xy☐ yxyx☐ xxyy☐ xxyxyxyy☐ xxyxyxyy☐ x☐ Inget av alternativen är rätt!

Fråga 17

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Syntax

Syntax 4. 3 poäng

De reguljära språken S1 och S2 beskrivs av

S1: $a^*b^*a^*b^*$ S2: $(ab^*)^*$

Kan snittet $S1 \cap S2$ beskrivas av ett regex?

☐ Nej☐ Ja, av $(ab^*)(ab^*)$ ☐ Ja, av $(ab^*)|ab^*ab^*|$ ☐ Ja, av $(ab^*)?(ab^*)?$ ☐ Ja, av $(ab^*ab^*)?$

Personnummer: 123456-7899

Namn: I, Carina

Avsluta provet

Personnummer: 123456-7899

Namn: I, Carina

Avsluta provet

Fråga 18

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Syntax

Syntax 5. 4 poäng

Betrakta följande grammatik med startsymbolen statement och slutsymboler med versaler:

```
statement ::= IF expr THEN statement
           | IF expr THEN statement ELSE statement
           | variable '=' expr
           | CONSTANT
           | expr OPERATOR expr
```

Vad är sant för grammatiken?

- ☐ Den är entydig.
- ☐ Den är flertydig.
- ☐ Den är LL(1).
- ☐ Den är kontextfri.
- ☐ Inget av alternativen är rätt.

Fråga 19

01	02	03	04	05	06	07	08	09	10
11	12	13	14	15	16	17	18	19	

Kategori: Syntax

Syntax 6. 4 poäng

Betrakta följande grammatik med startsymbolen x och slutsymbolerna A, I, O, U:

```
x ::= I x
      | O x I x
      | U x x
      | A
```

Anta att högerleden numreras från 1 till 4. I vilken ordning tillämpas högerleden av en

LL(1)-parser som läser strängen
OIAUUAIA

- ☐ 2 3 1 3 4 1 4
- ☐ 2 3 1 4 4 1 4
- ☐ 2 1 3 4 4 1 4
- ☐ 2 1 4 3 4 1 4
- ☐ Inget av alternativen är rätt.

Prolog

1 d

2 b

3 c

4 c

5 d

6 b

7 a

Haskell

1 bd

2 ab

3 bd

4 g

5 b

6 c

Syntax

1 b

2 ad

3 ace

4
↓
uAger

5 bd

6 d