

Försättsblad till skriftlig tentamen vid Linköpings universitet



Datum för tentamen	2017-10-28
Sal (1)	TER1(4)
Tid	14-18
Kurskod	TDDA69
Provkod	TENA
Kursnamn/benämning Provnamn/benämning	Data- och programstrukturer Tentamen
Institution	IDA
Antal uppgifter som ingår i tentamen	5
Jour/Kursansvarig Ange vem som besöker salen	Cyrille Berger
Telefon under skrivtiden	
Besöker salen ca klockan	
Kursadministratör/kontaktperson (namn + tfnr + mailaddress)	Anna Grabska Eklund, ankn. 2362, anna.grabska.eklund@liu.se
Tillåtna hjälpmedel	inga
Övrigt	
Antal exemplar i påsen	

TDDA69
Fall 2017
2017-10-27 14-18
Time Limit: 4 hours

Examiner: Cyrille Berger

This exam contains 3 pages (including this cover page) and 5 questions.
Total of points is 28p, the minimum for passing the exam is 14p, to get a four it is 18p and to get a five it is 23p.

No assistance.

Good luck!

1. (4 points) Programming paradigms.

(a) (3 points) Draw a diagram showing the relation between the following programming paradigms:

- First-order functional programming
- Functional programming
- Logic programming
- Imperative programming
- Sequential object-oriented programming

The relation between those programming paradigms could be (not all of them are necessarily useful, and some might appear several times in the diagram):

- +procedure
- +closure
- +cell(state)
- +unification
- +thread
- +search
- +port

The diagram should be a graph where the nodes are the programming paradigms and the edges are the relations.

(b) (1 point) Explain the main differences between weak typing and strong typing.

2. (2 points) Implement a function that print out all the permutation of a word using a recursion

```
1 def permutation(s):
2     pass # implement
3
4 permutation("eats")
```

This should print out:

```
1 teas
2 taes
3 tase
4 tsae
5 tsea
6 tesa
7 etas
8 eats
9 east
10 esat
11 esta
12 etsa
13 ates
14 aets
15 aest
16 aset
17 aste
18 atse
19 stea
20 seta
21 seat
22 saet
23 sate
24 stae
```

3. (8 points) Environment diagram.

Assume the expression below is evaluated in the order it is given.

```
1 function f(x)
2 {
3   return h(g)(x+1)(4, 5);
4 }
5 function g(x)
6 {
7   return function(y,z) { return z + (y * x); }
8 }
9 function h(f)
10 {
11   return function(x) { return f(x+3); }
12 }
13 f(5)
```

- (1 point) What will the result be?
- (3 points) Draw a diagram that captures what is going on according to the environment model of evaluation.
- (2 points) Mark the important structures and explain why, and in what order, they are created and (can be) removed.
- (2 points) Use the diagram to show the result of the evaluation.

4. (11 points) Virtual machines.
- (a) (2 points) What are the benefits of virtual machine over tree evaluation ? And what are the drawbacks?
 - (b) (3 points) How can you implement tail-call optimisation in a virtual machine?
 - (c) (1 point) Explain the purpose of a garbage collector.
 - (d) (2 points) Give an algorithm for memory allocation
 - (e) (3 points) Give an algorithm for garbage collection
5. (3 points) Logic Programming.
- (a) (1 point) Give the answer(s) to the following query:
 - 1 (fact (parent abraham barack))
 - 2 (fact (parent abraham clinton))
 - 3 (fact (parent delano herbert))
 - 4 (fact (parent fillmore delano))
 - 5 (fact (parent fillmore abraham))
 - 6 (fact (parent fillmore grover))
 - 7 (fact (grandparent (parent ?x ?y) (parent ?y ?z)))
 - 8
 - 9 (query (grandparent fillmore ?grandchild))
 - (b) (2 points) Explain how the query is executed.