

Exam: Models of Computation TDA183 – DIT310

Date: Dec 18, 2013, 8:30 – 12:30

Permitted aids: English-Swedish or English-other language dictionary.

Teacher: Bengt Nordstrom, phone 070 600 1546, Computer Science, University of Gothenburg and Chalmers

All solutions must be explained! It is not enough to just give a program without an explanation of why it works. The examination of the course consists of three parts: homework assignments, weekly exercises and this written exam (where each problem is worth 20 points). You have to have 100 points in total in order to pass the course.

Solutions to the exam will be available from the homepage of the course and it will be possible to discuss the grading on Thursday 23 Jan at 13:30 – 14:00 in Bengt Nordstrom's office.

Prove or disprove the following statements:

1. It is possible to write a function in Haskell (or some other language)

```
halt : Bool -> Bool
```

which returns `true` if the input terminates.

2. All closed expressions in lambda-calculus have a unique (up to alpha-conversion) normal form.
3. If an open expression in lambda-calculus has a normal form, then this normal form is open.
4. The full evaluation of a program in **X** always terminates if the program has a weak head normal form.
5. The set of all terminating Java-programs is enumerable.
6. The set of all total functions `Bool -> N` is enumerable.
7. It is possible to write a program in **PRF** which takes any number of arguments and always return 0. Here you must use the version of **PRF** which we have used in the course, so for instance the arity of `zero` is 0, the arity of `succ` is 1 and the arity of `proj(n,i)` is `n+1` if `i <= n`.

Good Luck!

Bengt