Domain-Specific Languages of Mathematics
Course codes: DAT326 / DIT983

Patrik Jansson

2024-08-27

Contact	Patrik Jansson, 072 985 2033.
Results	Announced within 15 workdays
Exam check	2023-09-20, 12.10-12.40 in EDIT 6128
Aids	One textbook of your choice (Domain-Specific Languages of Mathematics, or Beta - Mathematics Handbook, or Rudin, or Adams and Essex, or . . .). No printouts, no lecture notes, no notebooks, etc.
Grades	To pass you need **a minimum of 5p on each question (1 to 4)** and also reach these grade limits: 3: >=48p, 4: >=65p, 5: >=83p, max: 100p

Remember to write legibly. Good luck!

For reference: the learning outcomes. Some are tested by the hand-ins, some by the written exam.

- Knowledge and understanding
  - design and implement a DSL (Domain-Specific Language) for a new domain
  - organize areas of mathematics in DSL terms
  - explain main concepts of elementary real and complex analysis, algebra, and linear algebra
- Skills and abilities
  - develop adequate notation for mathematical concepts
  - perform calculational proofs
  - use power series for solving differential equations
  - use Laplace transforms for solving differential equations
- Judgement and approach
  - discuss and compare different software implementations of mathematical concepts

1. [25p] Algebraic structure: a DSL for "things"

   Consider the following (made up[1]) mathematical definition:

   $(X, start, grow, merge)$ is a **thing** with zero element *start*:

   $$merge\ start\ x == start == merge\ x\ start \qquad \text{-- for all } x \text{ in } X$$
   $$merge\ (grow\ x)\ (grow\ y) == grow\ (merge\ x\ y) \qquad \text{-- for all } x,\ y \text{ in } X$$

   (a) Define a type class *Thing* that corresponds to the "thing" structure.

   (b) Define a datatype $T\ v$ for the language of thing expressions (with variables of type v) and define a *Thing* instance for it. (These are expressions formed from applying the thing operations to the appropriate number of arguments, e.g., all the left hand sides and right hand sides of the above equations.)

   (c) Find and implement two other instances of the *Thing* class. Hint: look for one number-based and one string-based instance. Make sure the laws are satisfied.

   (d) Give a type signature for, and define, a general evaluator for $T\ v$ expressions on the basis of an assignment function.

   (e) Specialise the evaluator to the two *Thing* instances defined in (1c). Take three thing expressions of type $T\ String$, give the appropriate assignments and compute the results of evaluating, in each case, the three expressions.

2. [25p] **Laplace**

   Consider the following differential equation:

   $$f''/10 + f'/5 + f = 0, \quad f(0) = 2, \quad f'(0) = -2$$

   (a) [10p] Solve the equation assuming that f can be expressed by a power series *fs*, that is, use *integ* and the differential equation to express the relation between *fs*, *fs'*, *fs''*. What are the first three coefficients of *fs*? Explain how you compute them.

   (b) [15p] Solve the equation using the Laplace transform. You should need this formula (note that α can be a complex number) and the rules for linearity + derivative:

   $$\mathcal{L}\,(\lambda t.\, e^{\alpha * t})\, s = 1/(s - \alpha)$$

   Show that your solution does indeed satisfy the three requirements.

[1] Inspired by meet-semilattices with a monotone function.

3. [25pts] Adequate notation for mathematical concepts and proofs (or "flavours of continuity").

A formal definition of "$f : X \to \mathbb{R}$ is continuous" and "f is continuous at c" can be written as follows (using the helper predicate Q):

$$\begin{array}{ll} C(f) & = \forall\, c : X.\ Cat(f, c) \\ Cat(f, c) & = \forall\, \epsilon > 0.\ \exists\, \delta > 0.\ Q(f, c, \epsilon, \delta) \\ Q(f, c, \epsilon, \delta) & = \forall\, x : X.\ |x - c| < \delta \Rightarrow |f\ x - f\ c| < \epsilon \end{array}$$

By moving the existential quantifier outwards we can introduce the function $get\delta$ which computes the required δ from c and ϵ:

$$C'(f) = \exists\, get\delta : X \to \mathbb{R}_{>0} \to \mathbb{R}_{>0}.\ \forall\, c : X.\ \forall\, \epsilon > 0.\ Q(f, c, \epsilon, get\delta\ c\ \epsilon)$$

Now, consider this definition of *uniform continuity*:

Definition: Let $X \subseteq \mathbb{R}$. A function $f : X \to \mathbb{R}$ is *uniformly continuous* if for every $\epsilon > 0$, there exists $\delta > 0$ such that, for every x and y in the domain of f, if $|x - y| < \delta$, then $|f\ x - f\ y| < \epsilon$.

(a) [5pts] Write the definition of $UC(f) =$ "f is uniformly continuous" formally, using logical connectives and quantifiers. Try to use Q.

(b) [10pts] Transform $UC(f)$ into a new definition $UC'(f)$ by a transformation similar to the one from $C(f)$ to $C'(f)$. Explain the new function $new\delta$ introduced.

(c) [10pts] Prove that $\forall f : X \to \mathbb{R}.\ UC'(f) \Rightarrow C'(f)$. Explain your reasoning in terms of $get\delta$ and $new\delta$.

4. [25pts] Consider the following quote from Armitage & Griffiths, p. 158:

> Let $f : G \to \mathbb{R}^n$ be differentiable at $\boldsymbol{a} \in G \subset \mathbb{R}^n$. If $d_{\boldsymbol{a}} f : \mathbb{R}^n \to \mathbb{R}^n$ is a bijection, then there exist neighbourhoods $U_h(\boldsymbol{a}) \subset G$, $V_k(f(\boldsymbol{a})) \subset \mathbb{R}^n$, and a function $g : V_k \to U_h$ such that g is differentiable at $f(\boldsymbol{a})$; and for all $\boldsymbol{v} \in V_k$, $\boldsymbol{u} \in U_h$,
>
> $$g(f(\boldsymbol{u})) = \boldsymbol{u}, \qquad f(g(\boldsymbol{v})) = \boldsymbol{v}$$

Remark: In the context of the quote, $d_{\boldsymbol{a}} f$ denotes the differential of f in $\boldsymbol{a}$. The notation $U_r(x)$ is used to denote the set of points that are within a distance less than r to x ($r > 0$).

(a) What is the type of d?

(b) What are the types of h and k?

(c) What are the types of U and V?

(d) There seems to be an inconsistency between the types of U and V when they are introduced and their use in the following (in the types of $g, \boldsymbol{u}, \boldsymbol{v}$). What is the inconsistency? How can you fix it?