

1 О сертификации конструкций¹

Сертифицированная программа [1, стр. 7] — программа, снабженная конструктивным свидетельством своего соответствия требованиям, предъявляемым к ней спецификацией.

Конструктивность \neq алгоритмичность

Важно понимать, что термины «конструктивность» и «*алгоритмичность*» отнюдь не являются синонимами. *Конструктивность* объекта или процедуры несёт куда более широкий смысл, поскольку не утверждает ограниченности его свойств только формализуемыми в классической теории алгоритмов в духе Чёрча, Тьюринга, Поста и др. Так, природа объекта не обязана соответствовать подразумеваемым различными моделями ТА ограничениям типа фиксации исполнителя, дискретности, последовательности, детерминизма, отсутствия влияния «внешних» по отношению к системе «исполнитель-программа» факторов (взаимодействий, событий, исключений) и многого другого.

¹В принимаемом нами по умолчанию брауэровском интуиционистском (==конструктивистском) смысле.

Наши цели

Осенний семестр 2016

1. Разобрать основы генерации доказательств и сертифицированного программного кода в среде Coq [1, 3] (дополнительно [4, 5])
2. Знакомиться с конструктивизмом [Трулстра et al]
3. Углубить свои представления о решёточном подходе, пользуясь [Расёва-Сикорский (Гейтинг??)]

2016—2017 учебный год

- Выяснить перспективную применимость в интересующем нас контексте теории топосов [6, 7];
- Выявить потенциальные взаимосвязи рассматриваемых предметов с категорной стохастикой.
- Определить предпочтения относительно набора технологий *конструктивного* (в т. ч., машинного) представления математики.
- Получить выход на сжатие (устранение избыточности) данных — в особенности, естественно организованных во времени, таких, как сигналы, ряды, потоки. . . — в виду избранной парадигмы.

Магистратура

- Проверить действенность уни-(поли-?)валентной или иных реализаций теоретико-типового видения оснований математики при решении достаточно частных и значимых конкретных вопросов;
- Детально разработать такой вопрос в русле рассматриваемых тем;
- Представить результаты в магистерской диссертации;
- Определить дальнейшую траекторию

Литература

1. Chlipala A. Certified programming with dependent types : a pragmatic introduction to the Coq proof assistant / A. Chlipala, 2013. 424 с.
2. The Univalent Foundations Program. Homotopy Type Theory: Univalent Foundations of Mathematics. — Institute for Advanced Study : <https://homotopytypetheory.org/book>, 2013.
3. Coq Development Team. The Coq Proof Assistant Reference Manual [Электронный ресурс] : руководство пользователя — 2012. — Режим доступа: <https://coq.inria.fr/distrib/current/refman/index.html>.
4. Pierce B.C. [и др.]. Software Foundations // Version 4.0 (May, 2016) [Электронный ресурс]. URL: <http://www.cis.upenn.edu/~bcpierce/sf/current/index.html>.
5. Bertot Y., Cast?ran P. Interactive Theorem Proving and Program Development / Y. Bertot, P. Cast?ran, Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. URL: <http://www.cis.upenn.edu/~bcpierce/sf/current/index.html>.
6. Goldblatt R. Topoi : the categorial analysis of logic / R. Goldblatt, Dover Publications, 2006. 551 с.
7. Barr M., Wells C. TOPOSES, TRIPLES AND THEORIES 2005. URL: <http://www.tac.mta.ca/tac/reprints/articles/12/tr12.pdf>

8. Geuvers, H. Introduction to Type Theory / Herman Geuvers // Technical University Eindhoven, The Netherlands — 2008. — <http://www.cs.ru.nl/~herman/PUBS/IntroTT.pdf>.
9. Harper, R. Extensionality, intensionality, and Brouwer's dictum [Электронный ресурс] : электронная статья — 2012. — Режим доступа: <http://existentialtype.wordpress.com/2012/08/11/extensionality-intensionality-and-brouwers-dictum/>.