Introduction to Program Synthesis (SS 2025)

Chapter 1 - Introduction

Dr. rer. nat. Roman Kalkreuth

Chair for AI Methodology (**AIM**), Department of Computer Science,
RWTH Aachen University, Germany

- Discovery of algorithms with proofs → early in the history of **constructive mathematics**
  - ⤳ Explicitly avoids non-constructive proofs
  - ⤳ Constructive mathematics → proof should be algorithmic
  - ⤳ Proof design → based on examples
- Origin of the *proofs-as-programs* paradigm
- Follows **intuitionistic logic**
- Sometimes generally called constructive logic
- Does not allow non-constructive proofs

Zur Deutung der intuitionistischen Logik.

Von

A. Kolmogoroff in Moskau.

Figure: Kolmogorov's 1932 work on intuitionistic logic [Kol32]

Definition (Intuitionism)

- Fundamental idea $\rightarrow$ mathematics is a creation of the mind
- Truth of a mathematical statement $\rightarrow$ conceived via mental construction that proves it to be true
- Introduced by Brouwer
- Mathematical objects must be accessible to intuition
- Rejects non-constructive proofs

Definition (Intuitionistic logic)

- Form of logic studied and proposed by Gödel and Kolmogorov
- Formalises the only-constructive aspect of intuitionism

Proposition

There exist non-rational numbers a and b such that a^b is rational.

Proof.

We can proof that the above statement to be true by considering two cases:

- Case 1: $\sqrt{2}^{\sqrt{2}}$ is rational. Choose $a = \sqrt{2}$ and $b = \sqrt{2}$. Then a, b are irrational, and by assumption a^b is rational.
- Case 2: $\sqrt{2}^{\sqrt{2}}$ is irrational. Choose $a = \sqrt{2}^{\sqrt{2}}$ and $b = \sqrt{2}$. Then by assumption a, b are irrational and
$$a^b = \left(\sqrt{2}^{\sqrt{2}}\right)^{\sqrt{2}} = \left(\sqrt{2}\right)^{\sqrt{2}\cdot\sqrt{2}} = \left(\sqrt{2}\right)^2 = 2 \text{ is rational.}$$

$\square$

- Existence proof $\rightarrow$ non-constructive proof
  - $\rightsquigarrow$ Shows that a mathematical object exists without giving a concrete example
- Law of excluded middle
  - $\rightsquigarrow$ Means that there is no middle ground
- Minimalistic and simple constructive *proof*:
  - $\rightsquigarrow$ Set $a = \sqrt{2}$ and $b = \log_2 9$.
- Constructive proofs
  - $\rightsquigarrow$ Proofing the existence of a mathematical object by showing **how to create** the object

- **Curry-Howard correspondence [Cur34; How80; CFC59]**: Proofs can be considered programs and programs can be considered proofs
  - $\rightsquigarrow$ Isomorphism between **proof systems** and **computation models**
  - $\rightsquigarrow$ Proofs-as-programs paradigm
  - $\rightsquigarrow$ Isomorphism between **intuitionistic logic** and λ-**calculus**
  - $\rightsquigarrow$ λ-calculus $\rightarrow$ minimalistic formal system
  - $\rightsquigarrow$ Expression of algorithms as **compositions of functions**
- Transformation between theoretical and implementation level
- **Proof** $\cong$ **Program** $\rightarrow$ Formal systems
  - $\rightsquigarrow$ **Formal language** $\rightarrow$ constructed by a formal grammar
  - $\rightsquigarrow$ **Deductive system** $\rightarrow$ proof system
  - $\rightsquigarrow$ Natural deduction $\rightarrow$ logical reasoning
  - $\rightsquigarrow$ Frameworks for formal construction and reasoning

Lemma

If a and b are odd numbers, then $a+b$ is even.

Proof.

- Any odd number can be represented by $2n+1$ and by definition and any even number can be represented as $2n$ where n can be any integer
- Hence, by adding two odd numbers we obtain: $(2x+1)+(2y+1) = 2x+2y+2 = 2(x+y+1)$
- Since the sum of $x, y, 1$ is an integer we can define $n = x+y+1$
- Thus, the calculation leads then to an even number:

$$\begin{aligned} a+b &= (2x+1)+(2y+1) \\ &= 2x+2y+1 \\ &= 2(x+y+1) \\ &= 2n \end{aligned}$$

$\square$

- **Direct proof** that is algorithmic
- The **proof** can be considered as a **function** that has to be **implemented**
- We can derive and implement the definitions for odd and even numbers
  - $\rightsquigarrow$ Use of **compound data types** and **abstraction**

- **Inductive Proof** $\cong$ Iterative or recursive function
- For all integers $n \geq a$, a property $P(n)$ is true

Induction Proof	**Recursive Function**	**Iterative Function**
Proof $P(a)$	Base case definition	Initialisation statement definition
Assumption that $P(k)$ is true if $k \geq a \rightarrow$ inductive hypothesis	Recursive use of function $f(x)$. Assumption: It works with any value $k \geq a$	Condition holds for any $k \geq a$ and statement $S(k)$ yields a valid result
Show that if $P(k)$ is true then $P(k+1)$ is also true	Show that the result of $f(k)$ produces a valid result for $f(k+1)$	If the condition is true the expression leads to $k+1$

Proposition

$|\ \texttt{[None]} * n\ | = n$

Proof.

- Let $P(n)$ be the statement $|\ \texttt{[None]} * n\ | = n$
- $\exists n\, P(n)$
- $P(a) = P(0) \Rightarrow |\ [\,]\ | = |\emptyset| = 0$
- **Inductive hypothesis:** $P(k) = P(1) \Rightarrow |\ \texttt{[None]}\ | = 1 = |\ \texttt{[None]}\ | + |\ [\,]\ |$
- **Inductive step:** $P(k+1) = P(2) \Rightarrow |\ \texttt{[None, None]}\ | = |\ \texttt{[None]}\ | + |\ \texttt{[None]}\ | = 2$

$\square$

Proposition

$2^n > 2n,\ n > 2 \wedge n \in \mathbb{N}^+$

Proof.

- Let $P(n)$ be the statement $2^n > 2n$
- $P(a) = P(3) \Rightarrow 2^3 > 2 * 3 \Rightarrow 8 > 6$
- **Inductive hypothesis:** $P(k)$ holds for $2^k > 2k$
- **Inductive step:** $P(k+1) \Rightarrow 2^{k+1} = 2 * 2^k > 2 * 2k \Leftrightarrow 2 * 2^k > 2 * (k+1)$

$\square$

- Later more:
  - Recursion vs iteration
  - Inductive vs deductive approach
  - Tabulation vs memoisation $\rightsquigarrow$ dynamic programming

References I

[Kol32] A. Kolmogoroff. "Zur Deutung der intuitionistischen Logik". In: *Mathematische Zeitschrift* 35.1 (1932), pp. 58–65. DOI: 10.1007/BF01186549. URL: https://doi.org/10.1007/BF01186549.

[Cur34] H. B. Curry. "Functionality in Combinatory Logic". In: *Proceedings of the National Academy of Sciences* 20.11 (1934), pp. 584–590. DOI: 10.1073/pnas.20.11.584. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.20.11.584. URL: https://www.pnas.org/doi/abs/10.1073/pnas.20.11.584.

[How80] William Alvin Howard. "The Formulae-as-Types Notion of Construction". In: *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism*. Ed. by Haskell Curry et al. Academic Press, 1980.

[CFC59] Haskell B. Curry, Robert Feys, and William Craig. "Combinatory Logic, Volume I". In: *Philosophical Review* 68.4 (1959), pp. 548–550. DOI: 10.2307/2182503.