Todistuspeli

Kalle Kytölä
Aalto-yliopisto, Matematiikan ja systeemianalyysin laitos

On vaikea kuvitella helpompaa matemaattista käsitettä kuin luonnolliset luvut[1]: 0 on luonnollinen luku, 1 on luonnollinen luku, 2 on luonnollinen luku, 3 on luonnollinen luku, ja niin edelleen. Näennäisestä helppoudesta huolimatta olet epäilemättä törmännyt lukuteoreetikkojen vastalauseeseen. Luonnollisista luvuista on helppo esittää vaikeita kysymyksiä, joiden ratkaisut ovat vieneet ihmiskunnalta vuosisatoja (esim. Fermat'n suuri lause), tai kysymyksiä, joihin parhaatkaan matemaatikot eivät edelleenkään tiedä vastauksia. Siksi lukuteorian tutkimus jatkuukin aktiivisena.

Tarkoitukseni ei kuitenkaan ole toistaa tuota perusteltua vastalausetta helposti esitettävistä vaikeista kysymyksistä. Entä jos jopa helposti esitettävien helppojen kysymysten huolellinen ymmärtäminen on vaikeaa?

Jos pitkäveteisen tarinan sijaan mieluummin suoraan haastat itsesi koettamaan täsmällistä ymmärrystäsi helpoista kysymyksistä, niin artikkelin kohokohta on tämä linkki peliin, jossa pääset todistamaan luonnollisten lukujen aritmeettisia perusominaisuuksia:

`https://www.ma.imperial.ac.uk/~buzzard/xena/natural_number_game/`

Matemaattisista todistuksista

Matemaattiset tulokset ovat universaaleja totuuksia. Tuloksilla on täsmällinen merkitys ja niille esitettyjen aukottomien todistusten ansiosta saamme olla herttaisen yksimielisiä muun muassa siitä, että:

- $2 + 3 = 5$;
- ympyrän halkaisijan kahteen päätepisteeseen mistä tahansa muusta ympyrän pisteestä piirretyt suorat kohtaavat toisensa suorassa kulmassa;
- riippumattomien, samoin jakautuneiden, integroituvien satunnaismuuttujien jonon yhä pidempien äärellisten osajonojen keskiarvot suppenevat todennäköisyydellä 1 kohti kyseisten satunnaismuuttujien yhteistä odotusarvoa;

ja niin edelleen.

Tieteiden kuningattaren olemusta tavallisesti luonnehditaan tuohon tapaan: matematiikka koskee loogisella päättelyllä saavutettavia universaalisti voimassa olevia johtopäätöksiä. Ja tämä tosiaan on melko osuva kuvaus siitä, miten matematiikkaa on harjoitettu ainakin sitten Eukleides Aleksandrialaisen teoksen *Alkeet* (n. 300 eaa). Mutta luonnehdinta itse ei ole merkityksettään täsmällinen! Täsmennystä vaatisi vähintäänkin se, mitä tarkalleen ottaen tarkoitamme todistuksella —

[1] Merkitsemme tässä kirjoituksessa, kuten tavallista on, luonnollisten lukujen joukkoa $\mathbb{N} = \{0, 1, 2, \ldots\}$, kokonaislukujen joukkoa $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$, rationaalilukujen joukkoa $\mathbb{Q}$ ja reaalilukujen joukkoa $\mathbb{R}$.

ja vielä perustavammin, miten edes todistettavat väitteet tulisi muotoilla, jotta niillä olisi yksikäsitteinen merkitys, jota todistus sitten koskee. Matemaattinen logiikka käsittelee tuollaisia päänsärkyjä. Oma logiikan tuntemukseni on pinnallista, joten tyydyn kertomaan anekdootin ainoalta koskaan suorittamaltani logiikan kurssilta.

Ensimmäisen vuoden yliopistomatematiikan kurssilla *Logiikka I* esiteltiin propositiologiikan ja predikaattilogiikan täsmälliset kielet, joilla matemaattisia väitteitä yksikäsitteisesti muotoillaan. Kurssilla esiteltiin myös nk. "luonnollinen päättely" — tietynlainen propositio- ja predikaattilogiikan väitteiden todistusten kielioppi, jota hyväksyttävän loogisen todistuksen tulee noudattaa. Olin saanut houkuteltua seurakseni kurssille sosiaalipolitiikkaa ja filosofiaa opiskelevan entisen lukiokaverini. Kun sitten kurssin edetessä osoittautui, että väitteen "kyllä tai ei" (tarkemmin sanottuna: $P \vee \neg P$, missä P on mielivaltainen propositio) päättelemiseen kului paljon paperia ja aikaa, olin lievästi sanottuna turhautunut. Valtiotieteilijäystäväni sen sijaan tuumi, että luonnollinen päättely muistuttaa hieman tietokonepeliä, jossa yhä monimutkaisempien väitteiden todistaminen vastaa pelin yhä haastavampien tasojen läpäisemistä.

Formaaleista todistuksista

Ensimmäisen vuoden yliopisto-opiskelija pystyy pakon edessä tarkistamaan, noudattaako paperille raapustettu päättely logiikan sääntöjä viimeistä pilkkuakin myöten. Mutta eikö opiskelijaa sopivampi tällaiseen hanttihommaan olisikin tietokone? Siitä on kyse matematiikan formalisoinnissa: matemaattiset väitteet sekä niiden todistukset kirjoitetaan tietokoneen ymmärtämällä formaalilla kielellä ja kone tarkistaa sen, että todistus etenee logiikan sääntöjen mukaisesti ja väitetty tulos tulee näin todistetuksi.

Tarkoitukseen sopivia formaaleja kieliä on useita, mutta käytän esimerkkinä kieltä nimeltä *Lean*.

Karsittu esimerkki

Esimerkin paikan ansaitkoon lause numero 117 Eukleideen *Alkeet* -teoksen kirjassa X. Lauseen väite on, että luku $\sqrt{2}$ ei ole rationaalinen (toisin sanoen se on irrationaalinen). Klassinen todistus on lyhyt ja olet ehkä jo nähnytkin sen. Muistutetaan se kuitenkin ensin mieleen suurpiirteisesti ja katsotaan sitten (hyvin) yksityiskohtaisesti sen erästä päävaihetta.

Todistus on epäsuora, vastaoletukseen perustuva päättely. Vastaoletuksesta, että $\sqrt{2}$ olisikin rationaalinen, seuraisi, että se voidaan lausua muodossa $\sqrt{2} = \frac{n}{m}$ joillakin luonnollisilla luvuilla n ja m, joista $m \neq 0$ ja joilla ei ole yhteisiä alkutekijöitä (rationaaliluku on supistetussa muodossa). Neliöjuuren määrittelevän ominaisuuden mukaan silloin on

$$2 = \left(\frac{n}{m}\right)^2. \tag{1}$$

Todistuksen ydin on johtaa tästä lähtökohdasta ristiriita näyttämällä, että silloin sekä n että m ovat välttämättä parillisia, joten vastoin vastaoletusta niillä on yhteinen tekijä 2.

Keskittykäämme ainoastaan tuohon todistuksen ydinkohtaan, ja siitäkin vain osaan. Yhtälön (1) pienellä uudelleenjärjestelyllä saamme $2m^2 = n^2$, ja tavoitteenamme on ensin päätellä luvun n parillisuus tästä tiedosta. Vasen puoli $2m^2$ on selvästi parillinen, joten niin on oltava oikeankin puolen n^2, ja haluaisimme varmaankin osoittaa implikaation (seuraussuhteen)

$$n^2 \text{ on parillinen} \implies n \text{ on parillinen}. \tag{2}$$

Leanilla tämä implikaatio kirjoitetaan seuraavasti:

```
even (n^2) → even n.
```

Implikaation (2) todistamiseksi on kuitenkin paras edetä jälleen epäsuorasti ja ensin näyttää implikaatio

$$n \text{ on pariton} \implies n^2 \text{ on pariton}, \tag{3}$$

joka Leanilla kirjoitetaan seuraavasti:

```
odd n → odd (n^2).
```

Tämän aputuloksen eli lemman (3) formaaliksi todistukseksi Leanilla kelpaa seuraava koodinpätkä, jonka rivit nimeän kirjaimilla myöhempää kommentointia varten:

```
lemma parittoman_nelio_pariton
  {n : ℕ} :
  odd n → odd (n^2) :=
begin
  intro n_pariton,                    -- (A)
  unfold odd at *,                    -- (B)
  cases n_pariton with k n_esitys,    -- (C)
  rw n_esitys,                        -- (D)
  use 2 * k^2 + 2 * k,                -- (E)
  ring,                               -- (F)
end
```

Mitä tässä tapahtui? Ilmeisesti todistus alkaa taikasanalla `begin` ja päättyy vastaavasti `end`, mutta katsotaanpa niiden väliin jääviä rivejä yksitellen:

(A): Todistettavana on implikaatio (3), joten oletetaan implikaation vasen puoli (tässä tapauksessa että n on pariton) ja otetaan tavoitteeksi päätellä implikaation oikea puoli (tässä tapauksessa että n^2 on pariton). Tätä päättelysääntöä kutsutaan logiikassa *implikaation tuonniksi* ("implication introduction") ja Leanissa sen hoitaa komento `intro`. Näin käyttöön saatavalle hypoteesille annamme koodissamme nimen `n_pariton` — tämä on kuten mikä tahansa muuttujan nimi ohjelmointikielessä.

(B): Tässä vaiheessa on tärkeää tietää, mitä täsmälleen ottaen tarkoittaa se, että n (tai n^2) on pariton. Komentoa `unfold` käyttäen avaamme määritelmän termille `odd`, sekä hypoteesissamme `n_pariton` että tavoitteessamme. Painetusta koodista avaus ei valitettavasti näy päällepäin — koodatessa näkyy. Osoittautuu, että Leanissa täsmällinen määritelmä vaikkapa n:n parittomuudelle on, että $n = 2k + 1$ jollakin luonnollisella luvulla k. Siksi `n_pariton` tulee avatuksi muotoon

```
∃ (k : ℕ), n = 2 * k + 1.
```

Samaan tapaan avautuu tavoitteemme n^2:n parittomuudesta, koska pyysimme avausta kaikkialla, `at *`.

(C): Hypoteesimme `n_pariton` on nyt eksistenssikvanttorilla ∃ alkava propositio, johon soveltuu eksistenssikvanttorin eliminointisääntö ("existential quantifier elimination"). Se hoituu Leanin komennolla `cases`. Lopputuloksena saamme käyttöömme muuttujan k sekä uuden hypoteesin `n_esitys`, jonka mukaan $n = 2k + 1$.

(D): Tässä vaiheessa tavoitteemme on osoittaa n^2 parittomaksi, mikä parittomuuden määritelmän mukaan tarkoitti, että jollakin luonnollisella luvulla j pätee $n^2 = 2j + 1$. Hypoteesia `n_esitys` (joka siis sanoo $n = 2k + 1$) käyttäen voimme komennolla `rw` uudelleenkirjoittaa ("rewrite") tämän tavoitteen muotoon

```
∃ (j : ℕ), (2 * k + 1)^2 = 2 * j + 1.
```

(E): Lyhyellä binomikaavaan perustuvalla laskulla keksimme, että yllä oleva tavoitteemme ilmeisesti saavutettaisiin käyttämällä lukua $j = 2k^2 + 2k$. Formaalisti sovellamme eksistenssikvanttorin tuontisääntöä ("existential quantifier introduction") komennolla `use`.

(F): Todistus saataisiin loppuun tarkistamalla, että luku j yllä tosiaan toteuttaa halutun ominaisuuden $(2k + 1)^2 = 2j + 1$. Onneksi Leanin taktiikkakomento `ring` osaa binomikaavan luonnollisten lukujen (puoli)renkaassa[2] ℕ. Siksi Lean tässä vaiheessa tyytyväisenä ilmoittaa[3]:

goals accomplished

Näin selvitettyämme ensimmäisen aputuloksen (3) jatkakaamme vielä väitteen (2) formalisoinnin verran. En avaa sen formaalia todistusta yhtä yksityiskohtaisesti, mutta kokoan pikaiset kommentit kustakin rivistä jälleen alle. Huomionarvoista on:

- Todistuksessa (toiseksi viimeisessä vaiheessa) käytetään yllä todistamaamme lemmaa `parittoman_nelio_pariton` eli aputulosta (3). Tähän tapaan uusia matemaattisia tuloksia rakennetaan ennestään tunnettujen pohjalle.
- Muutamassa vaiheessa käytetään Leanin matemaattisessa kirjastossa *mathlib*[4] olevia valmiita tuloksia.[5]

```
lemma parillinen_jos_nelio_parillinen
  {n : ℕ} :
  even (n^2) → even n :=
begin
  intro parillinen_nelio,              -- (A)
  by_contradiction vastaoletus,        -- (B)
  have n_pariton : odd n,              -- (C)
    from nat.odd_iff_not_even.mpr vastaoletus,
  have pariton_nelio : odd (n^2),      -- (D)
    from parittoman_nelio_pariton n_pariton,
  exact nat.odd_iff_not_even.mp pariton_nelio
    parillinen_nelio,    -- (E)
end
```

Vaiheissa tapahtuu seuraavaa:

(A): Oletetaan implikaation vasen puoli eli että neliö n^2 on parillinen.

(B): Tehdään vastaoletus, että luku n ei ole parillinen.

(C): Silloin n on pariton. Perustelu: Luku on pariton jos (ja vain jos) se ei ole parillinen.

(D): Silloin myös neliö n^2 on pariton. Perustelu: Jo todistamamme lemma sekä tieto, että n on pariton.

(E): Ristiriita seuraa siitä, että luku on pariton (jos ja) vain jos se ei ole parillinen, mutta n^2 olisi yllä olevien mukaan sekä pariton että parillinen. Siis vastaoletus on väärä ja n oli välttämättä parillinen.

goals accomplished

Mutkia matkassa

Edellisten esimerkkien lemmojen tarkoitus oli havainnollistaa, millaista yksityiskohtien tasoa formaali looginen argumentti vaatii. Nämä lemmat muodostavat periaatteessa olennaisen osan luvun $\sqrt{2}$ irrationaalisuustodistuksen formalisointia; Lemmaa `parillinen_jos_nelio_parillinen` käyttäen yhtälöstä

[2] Itse asiassa `ring`-taktiikka osaa mitä tahansa mekaanisia renkaiden laskutoimituksia niin kompleksiluvuilla, matriiseilla kuin polynomeillakin — ylipäänsä missä tahansa renkaissa. Tietokone voi siis myös automatisoida päättelystä rutiininomaisia osia.

[3] Perinteisemmin todistuksen päättyminen voitaisiin ilmaista latinaksi *quod erat demonstrandum*, mikä on käännös Eukleideen käyttämästä kreikankielisestä ilmaisusta. Vielä tavallisempia ovat lyhenne *QED* tai ainoastaan symboli □. Yhtä kaikki, tässä kohtaa on syytä olla tyytyväinen.

[4] Varsinainen kirjasto on osoitteessa `https://github.com/leanprover-community/mathlib` ja paljon lisätietoa siitä löytyy projektin kotisivulta `https://leanprover-community.github.io/`.

[5] Toki $\sqrt{2}$:n irrationaalisuus olisi löytynyt kirjastosta sellaisenaan, mutta kirjaston käyttäminen siihen suoraan olisi tyystin vesittänyt tämän päättelyharjoituksemme!

$2m^2 = n^2$ (jonka perusteella n^2 on ilmeisesti parillinen) esimerkiksi pääteltäisiin, että n on parillinen.

Jos yrität itse näitä käyttäen kirjoittaa luvun $\sqrt{2}$ formaalin irrationaalisuustodistuksen loppuun asti Leanilla, vaikkapa muodossa

$$\forall \texttt{ (q : } \mathbb{Q}\texttt{), q\^{}2} \neq \texttt{2},$$

törmäät kuitenkin vielä muutamiin ehkä odottamattomiin hankaluuksiin.

Ensinnäkin, rationaaliluvut $q \in \mathbb{Q}$ Leanissa ovat (matemaattisesti oikein järkeenkäypän) määritelmän mukaan muotoa $\frac{n}{m}$, missä nimittäjä m on nollasta eroava luonnollinen luku, $m \in \mathbb{N}$, $m \neq 0$, ja osoittaja n on kokonaisluku $n \in \mathbb{Z}$, vieläpä niin että näiden suurin yhteinen tekijä on 1 (eli rationaaliluvut on jo Leanin määritelmän mukaan valittu esitettäväksi supistetussa muodossa). Mutta hupsis! Lemmamme yllä oli todistettu oletuksella, että myös n on *luonnollinen* luku. No eipä hätää, korvataan kohta $\{\texttt{n} : \mathbb{N}\}$ muotoon $\{\texttt{n} : \mathbb{Z}\}$, ja ainoa mitä aiemmissa todistuksissa tarvitsee muuttaa on korvata jälkimmäisessä `nat.odd_iff_not_even` muotoon `int.odd_iff_not_even`.

Pari askelta eteen, muutama taakse. Yhtälö (1) oli rationaalilukuja koskeva. Rationaalilukujen kuntaominaisuuksia käyttäen muutamalla sievennysvaiheella siitä tosiaan saadaan Leanissakin yhtälö $2m^2 = n^2$. Mutta tämä yhtälö on silloin johdettu *rationaaliluvuille* $2m^2$ ja n^2. Parillisuus ja parittomuus ovat kokonaislukujen ominaisuuksia, joten niistä puhumista varten molemmat puolet pitää ensin ymmärtää kokonaislukuina. Ja ovathan myös n ja m kokonaislukuja. Paitsi, että m oli luonnollinen luku. No, eihän tässä pitäisi olla mitään ongelmaa, koska jokainen luonnollinen luku on kokonaisluku, jokainen kokonaisluku on rationaaliluku ja jokainen rationaaliluku on reaaliluku (ja jokainen reaaliluku on vieläpä kompleksiluku), eli on voimassa osajoukkorelaatiot

$$\mathbb{N} \subseteq \mathbb{Z} \subseteq \mathbb{Q} \subseteq \mathbb{R}$$

(ja $\cdots \subseteq \mathbb{C}$). Leanissakin vastaava on kyllä jossakin määrin automaattista: tyyppimuunnokset (“coercion”)

$$\begin{aligned} \texttt{coe} &: \mathbb{N} \to \mathbb{Z} \\ \texttt{coe} &: \mathbb{Z} \to \mathbb{Q} \\ \texttt{coe} &: \mathbb{Q} \to \mathbb{R} \\ (\text{sekä } \texttt{coe} &: \mathbb{R} \to \mathbb{C}) \end{aligned}$$

hoituvat automaattisesti aina, kun Lean huomaa, että on esimerkiksi yritetty kirjoittaa kokonaisluku paikkaan, johon tarvittaisiin rationaaliluku.

Mutta lisäksi huolellisessa argumentaatiossa pitää kyllä käyttää havaintoa, että kokonaislukujen $2 \in \mathbb{Z}$ ja $m^2 \in \mathbb{Z}$ tulo $2m^2 \in \mathbb{Z}$ on sellainen, että sen tyyppimuunnos rationaaliluvuksi $(2m^2) \in \mathbb{Q}$ on sama kuin tyyppimuunnettujen lukujen $2 \in \mathbb{Q}$ ja $m^2 \in \mathbb{Q}$ tulo rationaalilukujen kunnassa, $2m^2 \in \mathbb{Q}$. Matemaattinen notaatiommekin

$$(2m^2) = 2m^2$$

ilman varta vasten merkitsemiäni sulkeita yrittäisi lakaista maton alle sen, että yhtälön vasemmalla puolella käytetään kokonaislukujen kertolaskua ennen tyyppimuunnosta, kun taas oikealla puolella tyyppimuunnos suoritetaan ensin ja sen jälkeen käytetään rationaalilukujen kertolaskua. Matemaatikoille tutummin asian ytimen voi ilmaista niin, että diagrammi

$$\begin{array}{ccc} \mathbb{Z} \times \mathbb{Z} & \xrightarrow{\mathbb{Z}\text{:n tulo}} & \mathbb{Z} \\ {\scriptstyle \text{coe}\times\text{coe}}\downarrow & & \downarrow{\scriptstyle \text{coe}} \\ \mathbb{Q} \times \mathbb{Q} & \xrightarrow[\mathbb{Q}\text{:n tulo}]{} & \mathbb{Q} \end{array} \tag{4}$$

kommutoi. Oletko koskaan hämmästellyt tai kummastellut diagrammiin (4) tiivistettyä ihmeellistä asiaa?

Lisäksi tyyppimuunnosten suunnan kääntäminen on vielä hankalampaa; jokaista rationaalilukua ei noin vain muunnetakaan kokonaisluvuksi tai luonnolliseksi luvuksi. Siksi sellaiselle rationaalilukuja koskevalle yhtälölle, jonka molemmat puolet ovat tyyppimuunnettuja kokonaislukuja, täytyy aidosti tehdä jotakin, jos halutaan vastaava kokonaislukuja koskeva yhtälö (olennaista tässä on tyyppimuunnosten injektiivisyys).

Niin omituiselta kuin se meille ihmisille vaikuttaakin, tällaiset pohdinnat täytyy jossakin vaiheessa perin pohjin selittää tietokoneelle. Kaikkia sellaisia tuskin tullaan koskaan saamaan erityisen kattavasti automatisoitua. Ihmismatemaatikot nimittäin hyppelehtivät tyyppimuunnoksia käyttäen kontekstista toiseen kiinnittämättä asiaan juurikaan huomiota. Kukaan ei hätkähdä sitä, että jokainen jatkuva funktio on myös funktio, tai että jokainen metrinen avaruus on myös topologinen avaruus, tai että jokainen rengas on myös vaihdannainen additiivinen ryhmä sekä multiplikatiivinen monoidi jne., eikä sitä, että monet näitä koskevat operaatiot muodostavat lisäksi kommutoivia diagrammeja tyyppimuunnosten kanssa. Havahduttuamme kiinnittämään asiaan huomiota vaikuttaa lähes ällistyttävältä, ettei matemaatikkojen suurpiirteisyys johda virheisiin päättelyssä kovinkaan usein.

Omia kokemuksiani

Ensimmäisenä omatoimisena formalisointiharjoituksenani päätin todistaa Leanilla itselleni hyvin tutun lauseen: nk. portmanteau-lauseen todennäköisyysmittojen heikon suppenemisen yhtäpitävistä karakterisaatioista. Olen opettanut sitä usein kursseillani. Luennoilla käytän sen todistukseen tyypillisesti puolisen tuntia. Saman opettaminen tietokoneelleni vaati minul-

ta yli 4000 koodiriviä ja hyvin hyvin monta viikonloppuiltaa. Erityisen hankalat kohdat liittyivät toisinaan vaiheisiin, jotka ovat matemaattisestikin mutkikkaampia, mutta usein myös vaiheisiin, joissa en (ihmis)matemaatikkona alunperin edes huomannut olevan mitään vaiheita.

Mutta miksi?

Vaikuttaa takkuiselta — miksi matematiikan formalisointia siis tehdään?

Useille ylivoimaisesti tärkein syy on, että formalisointi on hauskaa. Äärimmäisen hauskaa! Oletko sattumalta harrastanut shakkia tai gota, ratkonut Rubikin kuutiota, tai pelannut muita älypelejä? En hetkeäkään ihmettele *miksi*! Vanha lukiokaverini ymmärsi suhtautua päättelyihin jo logiikan fuksikurssilla pelinä.

Hauskuuden lisäksi formalisoinnille on toki myös muita, ehkä vakavampia syitä.

Verrattuna perinteiseen kirjoitettuun matemaattiseen todistukseen, melko ilmeinen etu tietokoneen tarkastamalla formaalilla todistuksella on virheettömyys. Vaikka matematiikka periaatteessa on loogisen täsmällistä, sitä tekevät ihmiset, jotka eivät ole erehtymättömiä[6]. Tieteen historiasta löytää lukuisia kiehtovia tarinoita virheistä matematiikassa (lähtien vaikkapa Andrew Wilesin ensimmäisestä julkisesti esittämästä todistuksesta Fermat'n suurelle lauseelle), eivätkä virheet luotettavina pidetyissä matematiikan tutkimusartikkeleissa ja oppikirjoissa ole myöskään harvinaisia. Osa nykyaikaisista todistuksista on niin monimutkaisia, että parhaidenkin matemaatikkojen on hyvin vaikea luotettavasti pitää argumentteja järjestyksessä mielessään. Vuoden 2020 loppupuolella Peter Scholtze, yksi maailman ehdottomista kärkimatemaatikoista (Fieldsin mitali 2018), esitti haasteen erään oman teoreemansa formalisoimisesta. Haasteessa hän kuvaili avoimesti sitä, kuinka harva tuon teoreeman todistuksen oli todella tarkastanut, ja myös omia aiempia erehdyksiään sekä huoliaan mahdollisista virheistä kyseisessä todistuksessa. Lean-yhteisö tarttui haasteeseen. Yhteisponnistuksella (ja Scholtzen tuella) noin kymmenen hengen ydinporukka sai puolessa vuodessa todistuksen olennaisimman osan formalisoitua. Koko todistuksen on edelleen käynyt läpi vain kourallinen alan matemaatikkoja, mutta tietokoneen hyväksymä formalisointi tekee siitä yhden yksityiskohtaisimmin tarkistetuista nykymatematiikan tuloksista.

Hyvin käytännöllinen syy formaalien kielten kehittämiselle ja käytölle on ohjelmistojen ja laitteiden toiminnan tarkastus. Samaan tapaan kuin matemaattisten todistusten logiikka käydään yksityiskohtia myöten läpi, voidaan varmistaa myös se, että esimerkiksi kriittiset sovellukset on toteutettu toimimaan annettujen ehtojen mukaisesti. Ohjelmisto- ja laitekehityksessä käytetään täysin samoja formaaleja (ohjelmointi)kieliä kuin matematiikan formalisoinnissa.

Muitakin vakavia ja käytännöllisiä syitä matematiikan formalisoinnille on, mutta en usko, että mikään niistä todella tavoittaa sen vaikuttavuutta. Formalisointi on vähintäänkin matematiikan digitalisointia. Vertailukohtana voisi pitää musiikkia tai valokuvia — niiden digitalisointia edeltäneillä vinyylilevyillä tai filmirullilla oli oma hohtonsa, mutta vaikkapa Spotify'tä tai itseohjautuvissa autoissa tarvittavaa koneopittua kuvantunnistusta ei sellaisten varaan olisi rakennettu. Pelkkä sisällön välittäminen ja kopioiminen tai vaikkapa tiedon louhinta digitaalisesta aineistosta mahdollistaa varmasti aivan uusia käyttötapoja matematiikallekin. On hyvin vaikea ennustaa, mitä kaikkea siitä seuraa tulevaisuudessa.

Peli!

Jos olet valmis lähestymään formaaleja todistuksia pelinä, on erinomainen ensiaskel luonnollisten lukujen helppojen ominaisuuksien johtamiseen keskittyvä *Natural Number Game*

`https://www.ma.imperial.ac.uk/~buzzard/xena/natural_number_game/`,

joka ei vaadi Leanin asentamista omalle tietokoneelle, vaan toimii sellaisenaan verkossa.

Arvokkaista kommenteista kiitokset ansaitsevat:
Jukka Kohonen
Milo Orlich
Juha Ruokolainen
Vadim Weinstein

[6] Korjatkaa, jos olen väärässä.