

**Lessons from the  
PrimeNumberTheorem+  
Project  
And Real Analysis Game**

---

**Alex Kontorovich**

---

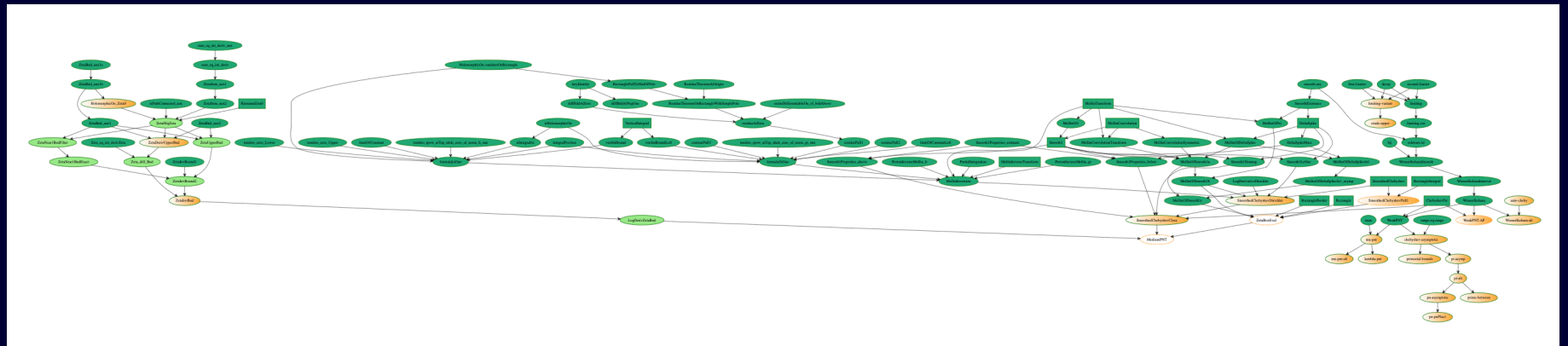
**RUTGERS UNIVERSITY / AWS**

# PrimeNumberTheorem+

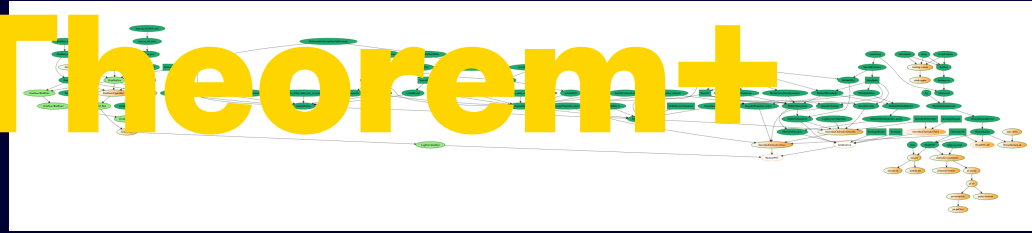
- Co-organized with Terry Tao
- Goal: Fermat will need Chebotarev Density Theorem. Special case of that is Dirichlet's theorem (primes in progressions). Didn't even have Prime Number Theorem in Mathlib. So let's get to work!
- Note: PNT has been formalized before, many times in fact.
- 2005: Avigad et al in Isabelle (Erdos-Selberg method);
- 2009: Harrison in HOL-light (Newman's proof);
- 2016: Carniero in Metamath (Erdos-Selberg);
- 2018: Eberl-Paulson in Isabelle (Newman)
- We will want to do in it a way that extends to much more general settings.
- Experiment in large scale collaboration.

# PrimeNumberTheorem+

- Experiment in large scale collaboration.
- Terry had just completed leading a formalization of the Polynomial Frieman-Rusza conjecture (Gowers-Green-Manners-Tao)
- Building on many other similar projects: Sphere Eversion (Patrick Massot), Liquid Tensor Experiment, etc, etc
- Organizational infrastructure: Github + Blueprint + Zulip



# PrimeNumberTheorem+

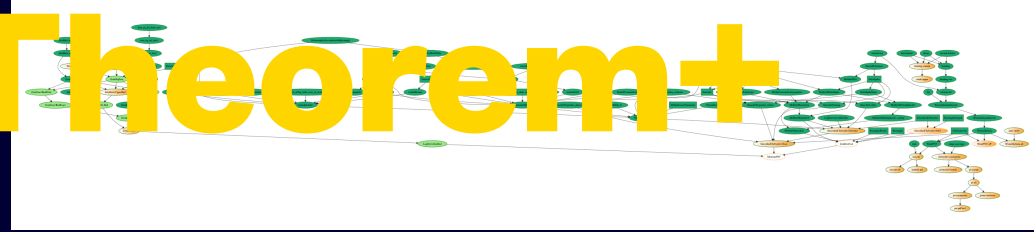


- Organizational infrastructure: Github + **Blueprint** + Zulip
- One “innovation” (**Ian Jauslin**+K): Interlace LaTeX with Lean, and scrape LaTeX before running Blueprint.
- This is amazing when coupled with Copilot, both to organize the writing, and autoformalize natural->formal language!

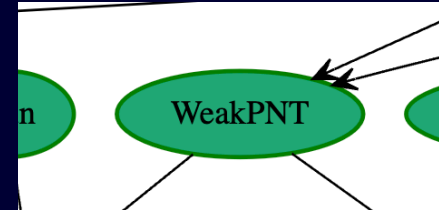
```
62 /-%%
63 \begin{lemma}[integrable_x_mul_Smooth1]\label{integrable_x_mul_Smooth1}
64 \lean{integrable_x_mul_Smooth1}\leanok
65 Fix a nonnegative, continuously differentiable function  $F$  on  $\mathbb{R}$ 
66 with support in  $[1/2, 2]$ , and total mass one,  $\int_{(0, \infty)} F(x)/x \, dx = 1$ . Then for any
67  $\epsilon > 0$ , the function
68  $x \mapsto x \cdot \widetilde{1_{[\epsilon, \infty)}}(x)$  is integrable on  $(0, \infty)$ .
69 \end{lemma}
70 %%-/
70 |lemma integrable_x_mul_Smooth1 {SmoothingF : ℝ → ℝ} (diffSmoothingF : ContDiff ℝ 1 SmoothingF)
70 | (SmoothingFpos : ∀ x, 0 ≤ SmoothingF x)
70 | (suppSmoothingF : Function.support SmoothingF ≤ Icc (1 / 2) 2) (mass_one : ∫ x in Ioi 0, Smo
70 | (ε : ℝ) (εpos : 0 < ε) :
70 | MeasureTheory.Integrable (fun x ↦ x * Smooth1 SmoothingF ε x) := by
70 | sorry
```

- Paul Nelson’s version in Emacs: sets TeX locally (VS extension??)

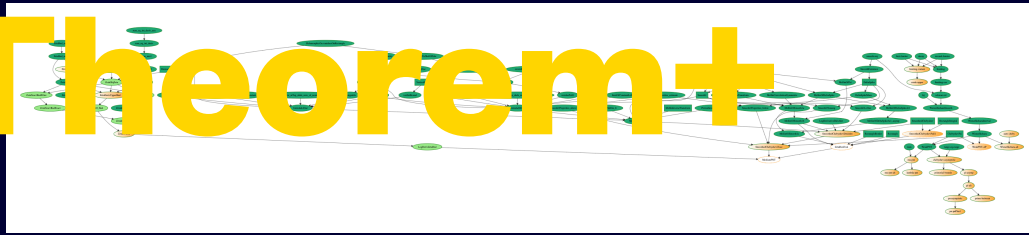
# PrimeNumberTheorem+



- So, how's it going?
- Opened to the “public” on Jan 31, 2024
- Proof of PNT completed on Apr 8, 2024
- Not at all a race; we did much more than that in the meantime and since
- Phase I of the project comprised three attacks on PNT:
  - (I) using “Fourier” methods, Wiener-Ikehara Tauberian theorem. Work of **Michael Stoll** already reduced PNT to this.
  - (II) developing Mellin transform API (**David Loeffler**), pulling infinite vertical contours past poles, picking up residues. And
  - (III) Getting a “classical” error savings of  $\exp(c(\log x)^{1/2})$  using Hadamard factorization (or local versions)



# PrimeNumberTheorem+



- Opened to the “public” on Jan 31, 2024
- Proof of PNT completed on Apr 8, 2024
- Phase I of the project comprised three attacks on PNT:
  - (I) using “Fourier” methods, Wiener-Ikehara Tauberian theorem. Work of **Michael Stoll** already reduced PNT to this.
  - (II) developing Mellin transform API (**David Loeffler**), pulling infinite vertical contours past poles, picking up residues. And
  - (III) Getting a “classical” error savings of  $\exp(c(\log x)^{1/2})$  using Hadamard factorization (or local versions)
- Mid-March 2024: **Shuhao Song**+**Bowen Yao** formalized (II) in Isabelle!
  - Built on top of much bigger Complex Analysis library...

# PrimeNumberTheorem+

- Phase I of the project comprised three attacks on PNT:
    - (I) using “Fourier” methods, Wiener-Ikehara Tauberian theorem. Work of **Michael Stoll** already reduced PNT to this.
    - (II) developing Mellin transform API (**David Loeffler**), pulling infinite vertical contours past poles, picking up residues. And
    - (III) Getting a “classical” error savings of  $\exp(c(\log x)^{1/2})$  using Hadamard factorization (or local versions)
  - (II) Was a great excuse to get lots of students and postdocs involved in Lean, culminating in June 2025 a Simons Foundation workshop <https://leanprover-community.github.io/blog/posts/simons-lean-workshop>
- Three weeks later: MediumPNT was done (saving  $\exp(c(\log x)^{1/10})$ )

# PrimeNumberTheorem+

- (II) Was a great excuse to get lots of students and postdocs involved in Lean, culminating in June 2025 a Simons Foundation workshop <https://leanprover-community.github.io/blog/posts/simons-lean-workshop>

Three weeks later: MediumPNT was done (saving  $\exp(c(\log x)^{1/10})$ )

- The solution was a mix of: (1) large scale collaborative human formalization, organized over a dedicated zulip channel, (2) copilot speeding up writing of LaTeX+Lean, (3) AlphaProof experiments during DeepMind's visit to IAS (where I was on sabbatical).
- (III) was announced in mid-Sept by Math Inc's Gauss solver, with Jared Lichtman iteratively writing out the blueprint in more and more detail, until the solver could auto-formalize the proof.



# PrimeNumberTheorem+

- The solution was a mix of: (1) large scale collaborative human formalization, organized over a dedicated zulip channel, (2) copilot speeding up writing of LaTeX+Lean, (3) AlphaProof experiments during DeepMind's visit to IAS (where I was on sabbatical).
- Compare AlphaProof solutions:
- AI invented its own language!
- Would not be easy to sort through this and figure out what's actually going on...

```
have ZetaBlowsUp : ∀f s in  $\mathcal{N}[\neq](1 : \mathbb{C})$ ,  $\| \zeta s \| \geq 1$  := by
  simp_all [Function.comp_def, eventually_nhdsWithin_iff, norm_eq_sqrt_real_inner]
  contrapose! h
  simp_all
  delta abs at*
  exfalso
  simp_rw [Metric.nhds_basis_ball.frequently_iff] at*
  choose! I A B using h
  choose a s using exists_seq_strictAnti_tendsto (0 :  $\mathbb{R}$ )
  apply ((isCompact_closedBall _ _).isSeqCompact fun and=>(A _ (s.2.1 and)).le.trans
    (s.2.2.bddAbove_range.some_mem (and, rfl))) .elim
  use fun and (a, H, S, M) => absurd (tendsto_nhds_unique M (tendsto_sub_nhds_zero_iff.
    1 ((squeeze_zero_norm fun and=>le_of_lt (A _ (s.2.1 _)) ) ) (s.2.2.comp S.
    tendsto_atTop)))) fun and=>?_
  norm_num [*, Function.comp_def] at M
  have := @riemannZeta_residue_one
  use one_ne_zero (tendsto_nhds_unique (this.comp (tendsto_nhdsWithin_iff.2 ( M, .
    of_forall (by norm_num [*])))) (squeeze_zero_norm ?_ ((M.sub_const 1).norm.trans
    (by rw [sub_self, norm_zero]))))
  use fun and =>.trans (norm_mul_le_of_le (le_rfl) (Complex.norm_def _>Real.
    sqrt_le_one.mpr (B ↑_ (s.2.1 ↑_)).right.le)) (by rw [mul_one])
```

# PrimeNumberTheorem+

- The solution was a mix of: (1) large scale collaborative human formalization, organized over a dedicated zulip channel, (2) copilot speeding up writing of LaTeX+Lean, (3) AlphaProof experiments during DeepMind's visit to IAS (where I was on sabbatical).
- Compare AlphaProof solutions:
- To Gauss (and GoedelProver, others...)
- Clear proof outline, complete with comments

```
lemma lem_m_rho_is_nat (R R1 : ℝ) (hR1_pos : 0 < R1) (hR1_lt_R : R1 < R) (f : ℂ → ℂ)
  (h_f_analytic : ∀ z ∈ Metric.closedBall (0 : ℂ) 1, AnalyticAt ℂ f z)
  (h_f_nonzero_at_zero : f 0 ≠ 0)
  (hR_lt_1 : R < 1) :
  ∀ (p : ℂ) (h_rho_in_KfR1 : p ∈ zerosetKfR R1 (by linarith) f),
  analyticOrderAt f p ≠ ∞ := by
  intro p h_rho_in_KfR1
  -- p lies in the closed ball of radius R1
  have hp_closed_R1 : p ∈ Metric.closedBall (0 : ℂ) R1 := h_rho_in_KfR1.1
  -- R1 < R and R < 1 implies R1 < 1
  have hR1_le_R : R1 ≤ R := by linarith
  have hR1_lt_one : R1 < 1 := by linarith
  -- Hence p ∈ ball 0 1
  have hp_ball1 : p ∈ Metric.ball (0 : ℂ) 1 := by
    have hdist_le : dist p (0 : ℂ) ≤ R1 := (Metric.mem_closedBall.mp hp_closed_R1)
    have hdist_lt : dist p (0 : ℂ) < 1 := by linarith
    simpa [Metric.mem_ball] using hdist_lt
  -- f is analytic at p
```

# PrimeNumberTheorem+

- Early phase: all human collaboration over zulip,
- Later: significant chunks can be done by automation (AlphaProof)
- Either way, project was a great excuse to do more analysis in Mathlib
- We didn't have Fourier inversion (now we do, [Sebastian Gouzel](#))
- We didn't have that Fourier transform of Schwartz function is Schwartz, now we do (Gouzel + K-Loeffler-[Macbeth](#) + [Beffara](#))
- We were also missing one of the least developed late undergrad / early grad areas of Mathlib (needed for lots of analytic number theory), namely: [Complex Analysis](#)

# Complex Analysis

- I asked Claude for the core results, and which were in Mathlib4:

## Complex Analysis Theorems in Mathlib4

### PROVED IN MATHLIB4 ✓

- Cauchy's integral theorem
- Cauchy's integral formula
- Liouville's theorem
- Fundamental theorem of algebra
- Maximum modulus principle
- Open mapping theorem
- Schwarz lemma
- Hadamard's theorem (three-lines theorem)

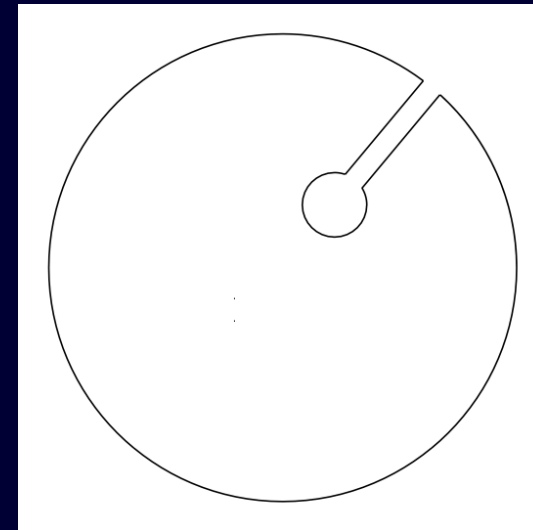
### NOT YET PROVED IN MATHLIB4 ✗

- Cauchy-Riemann equations
- Morera's theorem
- Identity theorem
- Argument principle
- Rouché's theorem
- Phragmén-Lindelöf principle
- Laurent series expansion
- Residue theorem
- Riemann mapping theorem
- Weierstrass factorization theorem
- Mittag-Leffler theorem
- Picard's theorems (little and great)
- Schwarz-Christoffel formula
- Complete Möbius transformations theory

V. Beffara

# Complex Analysis

- Residue Theorem:
- Don't need Jordan Curve Theorem, can do almost anything you want using explicit “Keyhole” contours
  - Do you really want to code up what this is, making sure to precompose every component with smoothing functions to stop at corners???
- Big Idea: Can do all the Complex Analysis we need just using Rectangles!
- Test (Ian Jauslin-K): holomorphic functions on disk have primitives



# Complex Analysis

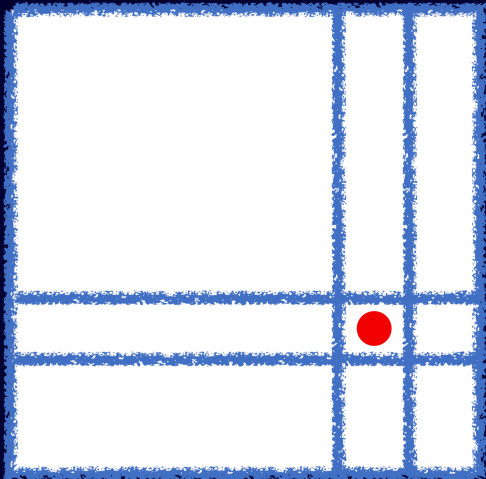
- Big Idea: Can do all the Complex Analysis we need just using Rectangles!
- We have Green's Theorem in Mathlib (Yury Kudryashov), so the integral of a holomorphic function over a rectangle is zero.

```
/--%%  
\begin{definition}\label{Rectangle}\lean{Rectangle}\leanok  
A Rectangle has corners  $z$  and  $w \in \mathbb{C}$ .  
\end{definition}  
%%-/  
/-- A `Rectangle` has corners `z` and `w`. -/  
def Rectangle (z w :  $\mathbb{C}$ ) : Set  $\mathbb{C}$  := [[z.re, w.re]] ×  $\mathbb{C}$  [[z.im, w.im]]
```

```
theorem HolomorphicOn.vanishesOnRectangle [CompleteSpace E] {U : Set  $\mathbb{C}$ }  
  (f_holo : HolomorphicOn f U) (hU : Rectangle z w  $\subseteq$  U) :  
  RectangleIntegral f z w = 0 :=  
  integral_boundary_rect_eq_zero_of_differentiableOn f z w (f_holo.mono hU)
```

# Complex Analysis

- Big Idea: Can do all the Complex Analysis we need just using Rectangles!
- We have Green's Theorem in Mathlib (Yury Kudryashov), so the integral of a holomorphic function over a rectangle is zero.
- And rectangles tile rectangles! (Not so with disks!)



- Chop big rectangle into 9 smaller rectangles
- 8 of them have integral zero!
- So you can zoom in as much as needed to a neighborhood of a pole

# Complex Analysis

- So you can zoom in as much as needed to a neighborhood of a pole



- Evaluating this boils down to:  $\int_{-\epsilon-i\epsilon}^{\epsilon+i\epsilon} \frac{ds}{s} = \int_{-\epsilon}^{\epsilon} \frac{dx}{x-i\epsilon} + \dots$

- Looks complicated! Need complex log??
- No, just add opposite sides together! Here is top + bottom:

$$\int_{-\epsilon}^{\epsilon} \frac{dx}{x-i\epsilon} - \int_{-\epsilon}^{\epsilon} \frac{dx}{x+i\epsilon} = 2i \int_{-1}^1 \frac{dx}{x^2+1} = 2i(\arctan 1 - \arctan(-1)) = \pi i$$

- Same with left side + right side.
- What about pulling contours?



# Complex Analysis

- What about pulling contours?
- Need things like Mellin transforms/inverses, e.g., Perron's formula:

$$\frac{1}{2\pi i} \int_{(2)} \frac{x^s}{s(s+1)} ds = \begin{cases} 0, & \text{if } x < 1 \\ 1 - \frac{1}{x}, & \text{if } x > 1 \end{cases}$$

- Human pf: if  $x < 1$ , pull contour to the left. Else pull right and pick up poles at  $s=0$  and  $s=-1$ .
- Lean:

theorem `Perron.formulaGtOne`

```
{x : ℝ} {σ : ℝ} (x_gt_one : 1 < x) (σ_pos : 0 < σ) :  
VerticalIntegral' (fun (s : ℂ) => ↑x ^ s / (s * (s + 1))) σ =  
1 - 1 / ↑x
```

- Pulling contours is adding/subtracting rectangles! + Limits

# Complex Analysis

- What about pulling contours?
- Need things like Mellin transforms/inverses, e.g., Perron's formula:

$$\frac{1}{2\pi i} \int_{(2)} \frac{x^s}{s(s+1)} ds = \begin{cases} 0, & \text{if } x < 1 \\ 1 - \frac{1}{x}, & \text{if } x > 1 \end{cases}$$

- Systematically use smoothing! (Standard in analytic number theory *except* for PNT!...) Compare to classical “unsmoothed” approach:

**Lemma 11.** *For any  $T > 0$  and  $b > 0$  we have*


$$\begin{cases} \left| \frac{1}{2\pi i} \int_{b-iT}^{b+iT} \frac{a^s}{s} ds - 1 \right| \leq \frac{a^b}{\pi T |\log a|}, & a > 1 \\ \left| \frac{1}{2\pi i} \int_{b-iT}^{b+iT} \frac{a^s}{s} ds \right| \leq \frac{a^b}{\pi T |\log a|}. & 0 < a < 1 \end{cases}$$

# PNT+

- Upshot:
  - Started massively collaborative;
  - Still very good mechanism for teaching newcomers formalization / Lean / Mathlib (slower, but speed isn't the point);
  - Newer paradigm: with API for Aristotle / AlphaProof / Gauss / Axiom / Aleph / GoedelProver / etc etc coming online, it's practically back to a single mathematician working "alone"!
- Lots of "novel" bits of pure mathematics discovered along the way; these should make their way into "standard" textbooks, as they actually give better / simpler / cleaner proofs!
- Does this "count" as research?

# Real Analysis, The Game!

- Lots of “novel” bits of pure mathematics discovered along the way; these should make their way into “standard” textbooks, as they actually give better / simpler / cleaner proofs!
- 44 worlds, 138 levels
- Proves all “greatest hits”:
  - “Bolzano-Weierstrass”
  - “Heine-Borel”

An Introduction to (Formal) Real Analysis


## Welcome to Real Analysis, The Game!


This course is currently being developed for Rutgers University Math 311H by [Alex Kontorovich](#). Please email [alex.kontorovich@rutgers.edu](mailto:alex.kontorovich@rutgers.edu) for suggestions/corrections, or better yet, post a PR/issue to <https://github.com/AlexKontorovich/RealAnalysisGame>.


For the main course website, go to: <https://alexkontorovich.github.io/2025F311H>.

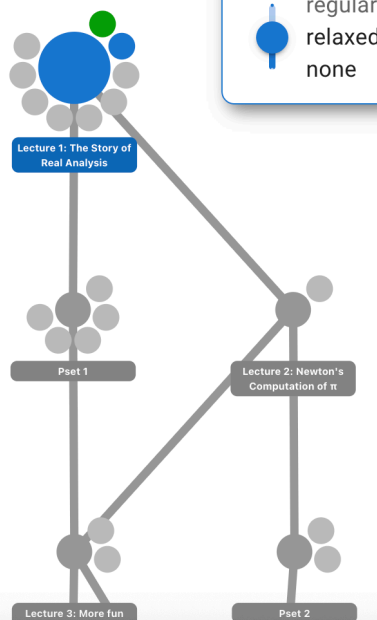
This course takes you through an Introduction to the Real Numbers, rigorous  $\epsilon - \delta$  Calculus, and basic Point-Set Topology. To get started, click on "Level 1: The Story of Real Analysis", and good luck!

**Rules** ?

 regular

 relaxed

 none



```
graph TD; L1[Lecture 1: The Story of Real Analysis] --> P1[Pset 1]; P1 --> L2[Lecture 2: Newton's Computation of pi]; P1 --> L3[Lecture 3: More fun]; P1 --> P2[Pset 2]; L2 --> P2; P2 --> P3[Pset 3]; P3 --> P4[Pset 4]; P4 --> P5[Pset 5]; P5 --> P6[Pset 6]; P6 --> P7[Pset 7]; P7 --> P8[Pset 8]; P8 --> P9[Pset 9]; P9 --> P10[Pset 10]; P10 --> P11[Pset 11]; P11 --> P12[Pset 12]; P12 --> P13[Pset 13]; P13 --> P14[Pset 14]; P14 --> P15[Pset 15]; P15 --> P16[Pset 16]; P16 --> P17[Pset 17]; P17 --> P18[Pset 18]; P18 --> P19[Pset 19]; P19 --> P20[Pset 20]; P20 --> P21[Pset 21]; P21 --> P22[Pset 22]; P22 --> P23[Pset 23]; P23 --> P24[Pset 24]; P24 --> P25[Pset 25]; P25 --> P26[Pset 26]; P26 --> P27[Pset 27]; P27 --> P28[Pset 28]; P28 --> P29[Pset 29]; P29 --> P30[Pset 30]; P30 --> P31[Pset 31]; P31 --> P32[Pset 32]; P32 --> P33[Pset 33]; P33 --> P34[Pset 34]; P34 --> P35[Pset 35]; P35 --> P36[Pset 36]; P36 --> P37[Pset 37]; P37 --> P38[Pset 38]; P38 --> P39[Pset 39]; P39 --> P40[Pset 40]; P40 --> P41[Pset 41]; P41 --> P42[Pset 42]; P42 --> P43[Pset 43]; P43 --> P44[Pset 44]; P44 --> P45[Pset 45]; P45 --> P46[Pset 46]; P46 --> P47[Pset 47]; P47 --> P48[Pset 48]; P48 --> P49[Pset 49]; P49 --> P50[Pset 50]; P50 --> P51[Pset 51]; P51 --> P52[Pset 52]; P52 --> P53[Pset 53]; P53 --> P54[Pset 54]; P54 --> P55[Pset 55]; P55 --> P56[Pset 56]; P56 --> P57[Pset 57]; P57 --> P58[Pset 58]; P58 --> P59[Pset 59]; P59 --> P60[Pset 60]; P60 --> P61[Pset 61]; P61 --> P62[Pset 62]; P62 --> P63[Pset 63]; P63 --> P64[Pset 64]; P64 --> P65[Pset 65]; P65 --> P66[Pset 66]; P66 --> P67[Pset 67]; P67 --> P68[Pset 68]; P68 --> P69[Pset 69]; P69 --> P70[Pset 70]; P70 --> P71[Pset 71]; P71 --> P72[Pset 72]; P72 --> P73[Pset 73]; P73 --> P74[Pset 74]; P74 --> P75[Pset 75]; P75 --> P76[Pset 76]; P76 --> P77[Pset 77]; P77 --> P78[Pset 78]; P78 --> P79[Pset 79]; P79 --> P80[Pset 80]; P80 --> P81[Pset 81]; P81 --> P82[Pset 82]; P82 --> P83[Pset 83]; P83 --> P84[Pset 84]; P84 --> P85[Pset 85]; P85 --> P86[Pset 86]; P86 --> P87[Pset 87]; P87 --> P88[Pset 88]; P88 --> P89[Pset 89]; P89 --> P90[Pset 90]; P90 --> P91[Pset 91]; P91 --> P92[Pset 92]; P92 --> P93[Pset 93]; P93 --> P94[Pset 94]; P94 --> P95[Pset 95]; P95 --> P96[Pset 96]; P96 --> P97[Pset 97]; P97 --> P98[Pset 98]; P98 --> P99[Pset 99]; P99 --> P100[Pset 100]; P100 --> P101[Pset 101]; P101 --> P102[Pset 102]; P102 --> P103[Pset 103]; P103 --> P104[Pset 104]; P104 --> P105[Pset 105]; P105 --> P106[Pset 106]; P106 --> P107[Pset 107]; P107 --> P108[Pset 108]; P108 --> P109[Pset 109]; P109 --> P110[Pset 110]; P110 --> P111[Pset 111]; P111 --> P112[Pset 112]; P112 --> P113[Pset 113]; P113 --> P114[Pset 114]; P114 --> P115[Pset 115]; P115 --> P116[Pset 116]; P116 --> P117[Pset 117]; P117 --> P118[Pset 118]; P118 --> P119[Pset 119]; P119 --> P120[Pset 120]; P120 --> P121[Pset 121]; P121 --> P122[Pset 122]; P122 --> P123[Pset 123]; P123 --> P124[Pset 124]; P124 --> P125[Pset 125]; P125 --> P126[Pset 126]; P126 --> P127[Pset 127]; P127 --> P128[Pset 128]; P128 --> P129[Pset 129]; P129 --> P130[Pset 130]; P130 --> P131[Pset 131]; P131 --> P132[Pset 132]; P132 --> P133[Pset 133]; P133 --> P134[Pset 134]; P134 --> P135[Pset 135]; P135 --> P136[Pset 136]; P136 --> P137[Pset 137]; P137 --> P138[Pset 138]; P138 --> P139[Pset 139]; P139 --> P140[Pset 140]; P140 --> P141[Pset 141]; P141 --> P142[Pset 142]; P142 --> P143[Pset 143]; P143 --> P144[Pset 144]; P144 --> P145[Pset 145]; P145 --> P146[Pset 146]; P146 --> P147[Pset 147]; P147 --> P148[Pset 148]; P148 --> P149[Pset 149]; P149 --> P150[Pset 150]; P150 --> P151[Pset 151]; P151 --> P152[Pset 152]; P152 --> P153[Pset 153]; P153 --> P154[Pset 154]; P154 --> P155[Pset 155]; P155 --> P156[Pset 156]; P156 --> P157[Pset 157]; P157 --> P158[Pset 158]; P158 --> P159[Pset 159]; P159 --> P160[Pset 160]; P160 --> P161[Pset 161]; P161 --> P162[Pset 162]; P162 --> P163[Pset 163]; P163 --> P164[Pset 164]; P164 --> P165[Pset 165]; P165 --> P166[Pset 166]; P166 --> P167[Pset 167]; P167 --> P168[Pset 168]; P168 --> P169[Pset 169]; P169 --> P170[Pset 170]; P170 --> P171[Pset 171]; P171 --> P172[Pset 172]; P172 --> P173[Pset 173]; P173 --> P174[Pset 174]; P174 --> P175[Pset 175]; P175 --> P176[Pset 176]; P176 --> P177[Pset 177]; P177 --> P178[Pset 178]; P178 --> P179[Pset 179]; P179 --> P180[Pset 180]; P180 --> P181[Pset 181]; P181 --> P182[Pset 182]; P182 --> P183[Pset 183]; P183 --> P184[Pset 184]; P184 --> P185[Pset 185]; P185 --> P186[Pset 186]; P186 --> P187[Pset 187]; P187 --> P188[Pset 188]; P188 --> P189[Pset 189]; P189 --> P190[Pset 190]; P190 --> P191[Pset 191]; P191 --> P192[Pset 192]; P192 --> P193[Pset 193]; P193 --> P194[Pset 194]; P194 --> P195[Pset 195]; P195 --> P196[Pset 196]; P196 --> P197[Pset 197]; P197 --> P198[Pset 198]; P198 --> P199[Pset 199]; P199 --> P200[Pset 200]; P200 --> P201[Pset 201]; P201 --> P202[Pset 202]; P202 --> P203[Pset 203]; P203 --> P204[Pset 204]; P204 --> P205[Pset 205]; P205 --> P206[Pset 206]; P206 --> P207[Pset 207]; P207 --> P208[Pset 208]; P208 --> P209[Pset 209]; P209 --> P210[Pset 210]; P210 --> P211[Pset 211]; P211 --> P212[Pset 212]; P212 --> P213[Pset 213]; P213 --> P214[Pset 214]; P214 --> P215[Pset 215]; P215 --> P216[Pset 216]; P216 --> P217[Pset 217]; P217 --> P218[Pset 218]; P218 --> P219[Pset 219]; P219 --> P220[Pset 220]; P220 --> P221[Pset 221]; P221 --> P222[Pset 222]; P222 --> P223[Pset 223]; P223 --> P224[Pset 224]; P224 --> P225[Pset 225]; P225 --> P226[Pset 226]; P226 --> P227[Pset 227]; P227 --> P228[Pset 228]; P228 --> P229[Pset 229]; P229 --> P230[Pset 230]; P230 --> P231[Pset 231]; P231 --> P232[Pset 232]; P232 --> P233[Pset 233]; P233 --> P234[Pset 234]; P234 --> P235[Pset 235]; P235 --> P236[Pset 236]; P236 --> P237[Pset 237]; P237 --> P238[Pset 238]; P238 --> P239[Pset 239]; P239 --> P240[Pset 240]; P240 --> P241[Pset 241]; P241 --> P242[Pset 242]; P242 --> P243[Pset 243]; P243 --> P244[Pset 244]; P244 --> P245[Pset 245]; P245 --> P246[Pset 246]; P246 --> P247[Pset 247]; P247 --> P248[Pset 248]; P248 --> P249[Pset 249]; P249 --> P250[Pset 250]; P250 --> P251[Pset 251]; P251 --> P252[Pset 252]; P252 --> P253[Pset 253]; P253 --> P254[Pset 254]; P254 --> P255[Pset 255]; P255 --> P256[Pset 256]; P256 --> P257[Pset 257]; P257 --> P258[Pset 258]; P258 --> P259[Pset 259]; P259 --> P260[Pset 260]; P260 --> P261[Pset 261]; P261 --> P262[Pset 262]; P262 --> P263[Pset 263]; P263 --> P264[Pset 264]; P264 --> P265[Pset 265]; P265 --> P266[Pset 266]; P266 --> P267[Pset 267]; P267 --> P268[Pset 268]; P268 --> P269[Pset 269]; P269 --> P270[Pset 270]; P270 --> P271[Pset 271]; P271 --> P272[Pset 272]; P272 --> P273[Pset 273]; P273 --> P274[Pset 274]; P274 --> P275[Pset 275]; P275 --> P276[Pset 276]; P276 --> P277[Pset 277]; P277 --> P278[Pset 278]; P278 --> P279[Pset 279]; P279 --> P280[Pset 280]; P280 --> P281[Pset 281]; P281 --> P282[Pset 282]; P282 --> P283[Pset 283]; P283 --> P284[Pset 284]; P284 --> P285[Pset 285]; P285 --> P286[Pset 286]; P286 --> P287[Pset 287]; P287 --> P288[Pset 288]; P288 --> P289[Pset 289]; P289 --> P290[Pset 290]; P290 --> P291[Pset 291]; P291 --> P292[Pset 292]; P292 --> P293[Pset 293]; P293 --> P294[Pset 294]; P294 --> P295[Pset 295]; P295 --> P296[Pset 296]; P296 --> P297[Pset 297]; P297 --> P298[Pset 298]; P298 --> P299[Pset 299]; P299 --> P300[Pset 300]; P300 --> P301[Pset 301]; P301 --> P302[Pset 302]; P302 --> P303[Pset 303]; P303 --> P304[Pset 304]; P304 --> P305[Pset 305]; P305 --> P306[Pset 306]; P306 --> P307[Pset 307]; P307 --> P308[Pset 308]; P308 --> P309[Pset 309]; P309 --> P310[Pset 310]; P310 --> P311[Pset 311]; P311 --> P312[Pset 312]; P312 --> P313[Pset 313]; P313 --> P314[Pset 314]; P314 --> P315[Pset 315]; P315 --> P316[Pset 316]; P316 --> P317[Pset 317]; P317 --> P318[Pset 318]; P318 --> P319[Pset 319]; P319 --> P320[Pset 320]; P320 --> P321[Pset 321]; P321 --> P322[Pset 322]; P322 --> P323[Pset 323]; P323 --> P324[Pset 324]; P324 --> P325[Pset 325]; P325 --> P326[Pset 326]; P326 --> P327[Pset 327]; P327 --> P328[Pset 328]; P328 --> P329[Pset 329]; P329 --> P330[Pset 330]; P330 --> P331[Pset 331]; P331 --> P332[Pset 332]; P332 --> P333[Pset 333]; P333 --> P334[Pset 334]; P334 --> P335[Pset 335]; P335 --> P336[Pset 336]; P336 --> P337[Pset 337]; P337 --> P338[Pset 338]; P338 --> P339[Pset 339]; P339 --> P340[Pset 340]; P340 --> P341[Pset 341]; P341 --> P342[Pset 342]; P342 --> P343[Pset 343]; P343 --> P344[Pset 344]; P344 --> P345[Pset 345]; P345 --> P346[Pset 346]; P346 --> P347[Pset 347]; P347 --> P348[Pset 348]; P348 --> P349[Pset 349]; P349 --> P350[Pset 350]; P350 --> P351[Pset 351]; P351 --> P352[Pset 352]; P352 --> P353[Pset 353]; P353 --> P354[Pset 354]; P354 --> P355[Pset 355]; P355 --> P356[Pset 356]; P356 --> P357[Pset 357]; P357 --> P358[Pset 358]; P358 --> P359[Pset 359]; P359 --> P360[Pset 360]; P360 --> P361[Pset 361]; P361 --> P362[Pset 362]; P362 --> P363[Pset 363]; P363 --> P364[Pset 364]; P364 --> P365[Pset 365]; P365 --> P366[Pset 366]; P366 --> P367[Pset 367]; P367 --> P368[Pset 368]; P368 --> P369[Pset 369]; P369 --> P370[Pset 370]; P370 --> P371[Pset 371]; P371 --> P372[Pset 372]; P372 --> P373[Pset 373]; P373 --> P374[Pset 374]; P374 --> P375[Pset 375]; P375 --> P376[Pset 376]; P376 --> P377[Pset 377]; P377 --> P378[Pset 378]; P378 --> P379[Pset 379]; P379 --> P380[Pset 380]; P380 --> P381[Pset 381]; P381 --> P382[Pset 382]; P382 --> P383[Pset 383]; P383 --> P384[Pset 384]; P384 --> P385[Pset 385]; P385 --> P386[Pset 386]; P386 --> P387[Pset 387]; P387 --> P388[Pset 388]; P388 --> P389[Pset 389]; P389 --> P390[Pset 390]; P390 --> P391[Pset 391]; P391 --> P392[Pset 392]; P392 --> P393[Pset 393]; P393 --> P394[Pset 394]; P394 --> P395[Pset 395]; P395 --> P396[Pset 396]; P396 --> P397[Pset 397]; P397 --> P398[Pset 398]; P398 --> P399[Pset 399]; P399 --> P400[Pset 400]; P400 --> P401[Pset 401]; P401 --> P402[Pset 402]; P402 --> P403[Pset 403]; P403 --> P404[Pset 404]; P404 --> P405[Pset 405]; P405 --> P406[Pset 406]; P406 --> P407[Pset 407]; P407 --> P408[Pset 408]; P408 --> P409[Pset 409]; P409 --> P410[Pset 410]; P410 --> P411[Pset 411]; P411 --> P412[Pset 412]; P412 --> P413[Pset 413]; P413 --> P414[Pset 414]; P414 --> P415[Pset 415]; P415 --> P416[Pset 416]; P416 --> P417[Pset 417]; P417 --> P418[Pset 418]; P418 --> P419[Pset 419]; P419 --> P420[Pset 420]; P420 --> P421[Pset 421]; P421 --> P422[Pset 422]; P422 --> P423[Pset 423]; P423 --> P424[Pset 424]; P424 --> P425[Pset 425]; P425 --> P426[Pset 426]; P426 --> P427[Pset 427]; P427 --> P428[Pset 428]; P428 --> P429[Pset 429]; P429 --> P430[Pset 430]; P430 --> P431[Pset 431]; P431 --> P432[Pset 432]; P432 --> P433[Pset 433]; P433 --> P434[Pset 434]; P434 --> P435[Pset 435]; P435 --> P436[Pset 436]; P436 --> P437[Pset 437]; P437 --> P438[Pset 438]; P438 --> P439[Pset 439]; P439 --> P440[Pset 440]; P440 --> P441[Pset 441]; P441 --> P442[Pset 442]; P442 --> P443[Pset 443]; P443 --> P444[Pset 444]; P444 --> P445[Pset 445]; P445 --> P446[Pset 446]; P446 --> P447[Pset 447]; P447 --> P448[Pset 448]; P448 --> P449[Pset 449]; P449 --> P450[Pset 450]; P450 --> P451[Pset 451]; P451 --> P452[Pset 452]; P452 --> P453[Pset 453]; P453 --> P454[Pset 454]; P454 --> P455[Pset 455]; P455 --> P456[Pset 456]; P456 --> P457[Pset 457]; P457 --> P458[Pset 458]; P458 --> P459[Pset 459]; P459 --> P460[Pset 460]; P460 --> P461[Pset 461]; P461 --> P462[Pset 462]; P462 --> P463[Pset 463]; P463 --> P464[Pset 464]; P464 --> P465[Pset 465]; P465 --> P466[Pset 466]; P466 --> P467[Pset 467]; P467 --> P468[Pset 468]; P468 --> P469[Pset 469]; P469 --> P470[Pset 470]; P470 --> P471[Pset 471]; P471 --> P472[Pset 472]; P472 --> P473[Pset 473]; P473 --> P474[Pset 474]; P474 --> P475[Pset 475]; P475 --> P476[Pset 476]; P476 --> P477[Pset 477]; P477 --> P478[Pset 478]; P478 --> P479[Pset 479]; P479 --> P480[Pset 480]; P480 --> P481[Pset 481]; P481 --> P482[Pset 482]; P482 --> P483[Pset 483]; P483 --> P484[Pset 484]; P484 --> P485[Pset 485]; P485 --> P486[Pset 486]; P486 --> P487[Pset 487]; P487 --> P488[Pset 488]; P488 --> P489[Pset 489]; P489 --> P490[Pset 490]; P490 --> P491[Pset 491]; P491 --> P492[Pset 492]; P492 --> P493[Pset 493]; P493 --> P494[Pset 494]; P494 --> P495[Pset 495]; P495 --> P496[Pset 496]; P496 --> P497[Pset 497]; P497 --> P498[Pset 498]; P498 --> P499[Pset 499]; P499 --> P500[Pset 500]; P500 --> P501[Pset 501]; P501 --> P502[Pset 502]; P502 --> P503[Pset 503]; P503 --> P504[Pset 504]; P504 --> P505[Pset 505]; P505 --> P506[Pset 506]; P506 --> P507[Pset 507]; P507 --> P508[Pset 508]; P508 --> P509[Pset 509]; P509 --> P510[Pset 510]; P510 --> P511[Pset 511]; P511 --> P512[Pset 512]; P512 --> P513[Pset 513]; P513 --> P514[Pset 514]; P514 --> P515[Pset 515]; P515 --> P516[Pset 516]; P516 --> P517[Pset 517]; P517 --> P518[Pset 518]; P518 --> P519[Pset 519]; P519 --> P520[Pset 520]; P520 --> P521[Pset 521]; P521 --> P522[Pset 522]; P522 --> P523[Pset 523]; P523 --> P524[Pset 524]; P524 --> P525[Pset 525]; P525 --> P526[Pset 526]; P526 --> P527[Pset 527]; P527 --> P528[Pset 528]; P528 --> P529[Pset 529]; P529 --> P530[Pset 530]; P530 --> P531[Pset 531]; P531 --> P532[Pset 532]; P532 --> P533[Pset 533]; P533 --> P534[Pset 534]; P534 --> P535[Pset 535]; P535 --> P536[Pset 536]; P536 --> P537[Pset 537]; P537 --> P538[Pset 538]; P538 --> P539[Pset 539]; P539 --> P540[Pset 540]; P540 --> P541[Pset 541]; P541 --> P542[Pset 542]; P542 --> P543[Pset 543]; P543 --> P544[Pset 544]; P544 --> P545[Pset 545]; P545 --> P546[Pset 546]; P546 --> P547[Pset 547]; P547 --> P548[Pset 548]; P548 --> P549[Pset 549]; P549 --> P550[Pset 550]; P550 --> P551[Pset 551]; P551 --> P552[Pset 552]; P552 --> P553[Pset 553]; P553 --> P554[Pset 554]; P554 --> P555[Pset 555]; P555 --> P556[Pset 556]; P556 --> P557[Pset 557]; P557 --> P558[Pset 558]; P558 --> P559[Pset 559]; P559 --> P560[Pset 560]; P560 --> P561[Pset 561]; P561 --> P562[Pset 562]; P562 --> P563[Pset 563]; P563 --> P564[Pset 564]; P564 --> P565[Pset 565]; P565 --> P566[Pset 566]; P566 --> P567[Pset 567]; P567 --> P568[Pset 568]; P568 --> P569[Pset 569]; P569 --> P570[Pset 570]; P570 --> P571[Pset 571]; P571 --> P572[Pset 572]; P572 --> P573[Pset 573]; P573 --> P574[Pset 574]; P574 --> P575[Pset 575]; P575 --> P576[Pset 576]; P576 --> P577[Pset 577]; P577 --> P578[Pset 578]; P578 --> P579[Pset 579]; P579 --> P580[Pset 580]; P580 --> P581[Pset 581]; P581 --> P582[Pset 582]; P582 --> P583[Pset 583]; P583 --> P584[Pset 584]; P584 --> P585[Pset 585]; P585 --> P586[Pset 586]; P586 --> P587[Pset 587]; P587 --> P588[Pset 588]; P588 --> P589[Pset 589]; P589 --> P590[Pset 590]; P590 --> P591[Pset 591]; P591 --> P592[Pset 592]; P592 --> P593[Pset 593]; P593 --> P594[Pset 594]; P594 --> P595[Pset 595]; P595 --> P596[Pset 596]; P596 --> P597[Pset 597]; P597 --> P598[Pset 598]; P598 --> P599[Pset 599]; P599 --> P600[Pset 600]; P600 --> P601[Pset 601]; P601 --> P602[Pset 602]; P602 --> P603[Pset 603]; P603 --> P604[Pset 604]; P604 --> P605[Pset 605]; P605 --> P606[Pset 606]; P606 --> P607[Pset 607]; P607 --> P608[Pset 608]; P608 --> P609[Pset 609]; P609 --> P610[Pset 610]; P610 --> P611[Pset 611]; P611 --> P612[Pset 612]; P612 --> P613[Pset 613]; P613 --> P614[Pset 614]; P614 --> P615[Pset 615]; P615 --> P616[Pset 616]; P616 --> P617[Pset 617]; P617 --> P618[Pset 618]; P618 --> P619[Pset 619]; P619 --> P620[Pset 620]; P620 --> P621[Pset 621]; P621 --> P622[Pset 622]; P622 --> P623[Pset 623]; P623 --> P624[Pset 624]; P624 --> P625[Pset 625]; P625 --> P626[Pset 626]; P626 --> P627[Pset 627]; P627 --> P628[Pset 6
```

---

# Real Analysis, The Game!

- “Bolzano-Weierstrass”
  - Classical statement: a bounded sequence of real numbers has a convergence subsequence.
  - Classical proof: Say bounded in  $[0,1]$ ; infinitely many points are in (at least) one of  $[0,1/2], [1/2,1]$ . Keep bisecting, keeping a branch where there are infinitely many. Done.
  - Not so easy (not impossible...) to formalize!

---

# Real Analysis, The Game!

- “Bolzano-Weierstrass”
  - Classical statement: a bounded sequence of real numbers has a convergence subsequence.
  - “Better” statement: a bounded sequence of real or *rational* (!) numbers has a subsequence that is Cauchy!
  - This can be proved *before* constructing the real numbers.
  - Proof is much more elementary: (1) Every sequence has a Monotone or Antitone subsequence (consider “Peaks” and whether there are infinitely many or not). (2) Bounded + Monotone/Antitone  $\Rightarrow$  Cauchy! Very easy proofs.

---

# Real Analysis, The Game!

- This can be proved *before* constructing the real numbers.
- Pedagogical philosophy: we are **not** building a house; we're thinking about **why** houses are built the way they are!
- When building a house, you pour the foundation, then frame everything, then run pipes/electric, only then do you put up the walls.
- But you *live* in the walls! So to *learn* about a house, you might make the walls first, since they're the most familiar. Then you realize you'd like running water, so you add plumbing. Walls start to fall, so you add framing. House sags, so you add foundation.

---

# Real Analysis, The Game!

- This can be proved *before* constructing the real numbers.
- But you *live* in the walls! So to *learn* about a house, you might make the walls first, since they're the most familiar. Then you realize you'd like running water, so you add plumbing. Walls start to fall, so you add framing. House sags, so you add foundation.
- You would not **live** in such a house, but you would learn a lot from the process, about *why* everything is being done the way it is!
- Same with The Game: ideas are only introduced at the moment they're necessary.



---

# Real Analysis, The Game!

- Same with The Game: ideas are only introduced at the moment they're necessary.
- Compare to standard Real Analysis texts: they feel like they're making a version of Mathlib:
  - Ch 1: Construct Reals (otherwise, how do you know what you're talking about?)
  - Ch 2/3: Sequences/Series
  - Ch 4: Topology in Reals (I thought this course was Analysis?)
  - Ch 5/6/7: Continuous Functions/Limits/Derivatives/Integrals

# Real Analysis, The Game!

- In The Game:
  - Ch 1/2: Sequences/Series
  - (Pretend you “know”  $\mathbb{R}$ !)
  - Ch 3: Construct Reals!
  - Ch 4/5: Continuous Functions/Limits/Derivatives/Integrals
  - Q: When does sequence of Riemann Sums converge? A: When  $f$  is uniformly continuous! When is that? When  $f$  is continuous on a compact domain. What’s that?
  - Ch 6: Topology in Reals.
  - Ch 7: Fancy things, interchange integrals+limits, etc.
- Compare to standard Real Analysis texts: they feel like they’re making a version of Mathlib:
  - Ch 1: Construct Reals (otherwise, how do you know what you’re talking about?)
  - Ch 2/3: Sequences/Series
  - Ch 4: Topology in Reals (I thought this course was Analysis?)
  - Ch 5/6/7: Continuous Functions/Limits/Derivatives/Integrals

# Real Analysis, The Game!

- In The Game:
  - “Heine-Borel”:

```
def IsCompact (S : Set ℝ) : Prop :=  
  ∀ (ι : Type) (xs : ι → ℝ) (rs : ι → ℝ), (∀ i, 0 < rs i)  
  → (S ⊆ ∪ i, Ball (xs i) (rs i)) →  
  ∃ (V : Finset ι), S ⊆ ∪ i ∈ V, Ball (xs i) (rs i)
```

- Compactness: any cover *by balls* has a finite subcover!
- Equivalent to open covers, but much easier to work with formally.

# Real Analysis, The Game!

- In The Game:
- “Heine-Borel”:

```
def IsCompact (S : Set ℝ) : Prop :=  
  ∀ (ι : Type) (xs : ι → ℝ) (rs : ι → ℝ), (∀ i, 0 < rs i)  
  → (S ⊆ ∪ i, Ball (xs i) (rs i)) →  
  ∃ (V : Finset ι), S ⊆ ∪ i ∈ V, Ball (xs i) (rs i)
```

```
Statement IsCompact_of_ClosedSubset {S T : Set ℝ} (hST : S ⊆ T)  
(hT : IsCompact T) (hS : IsClosed S) : IsCompact S := by  
  intro ι xs rs rpos hcover  
  change IsOpen Sc at hS  
  change ∀ x ∈ Sc, ∃ r > 0, Ball x r ⊆ Sc at hS  
  choose δs δspos hδs using hS  
  let Sbar : Set ℝ := Sc  
  let J : Type := Sbar  
  let U : Type := ι ⊕ J  
  let xs' : U → ℝ := fun i =>  
    match i with  
    | Sum.inl j => xs j  
    | Sum.inr x => x  
  let rs' : U → ℝ := fun i =>  
    match i with  
    | Sum.inl j => rs j  
    | Sum.inr x => δs x.1 x.2
```

- Easy: compact  $\Rightarrow$  bounded
- Medium: compact  $\Rightarrow$  closed
- Medium:  $[a, b] \Rightarrow$  compact
- Hardest: closed  $\subseteq$  cpt  $\Rightarrow$  cpt
- Q: why Types? Can't we just index everything by  $\mathbb{N}$ ? :)

---

# Real Analysis, The Game!

- In The Game:
  - Lots more such “innovations”.
  - (Riemann Rearrangement / Conditional Convergence Thm...)
- The real question is: do the students actually learn Real Analysis??
- Null hypothesis: Students with no prior exposure to Lean nor to the subject matter cannot effectively learn both.
- I must reject the null hypothesis...

# Real Analysis, The Game!

- I must reject the null hypothesis...

6. (10 pts) We say that a sequence  $a : \mathbb{N} \rightarrow \mathbb{R}$  diverges to infinity if it eventually exceeds any bound. More precisely:

```
def DivToInf (a:  $\mathbb{N} \rightarrow \mathbb{R}$ ) : Prop :=  $\forall M > 0, \exists N, \forall n \geq N, a\ n > M$ 
```

Give a formal proof of this fact, followed by a natural language statement (explaining what the theorem says) and natural language proof:

Objects:

$a\ b\ c : \mathbb{N} \rightarrow \mathbb{R}$

$L : \mathbb{R}$

Assumptions:

(ha : DivToInf a)  $\forall M > 0, \exists N, \forall n \geq N, a\ n > M$

(hb : SeqLim b L)  $\forall \epsilon > 0, \exists N, \forall n \geq N, |b\ n - L| < \epsilon$

(hL :  $L > 0$ )

(hc :  $\forall n, c\ n = a\ n * b\ n$ )

Goal: DivToInf c

hm:  $M > 0$

intro M hm  
specialize hb (1/2) (by bound) ✓  
specialize ha (2M/L) (by bound) ✓  
choose N1 hN1 using ha  
choose N2 hN2 using hb  
specialize hN1 (N1 hN1) (by bound)  
specialize hN2 (N2 hN2) (by bound)  
rewrite [abs\_lt] at hN2  
have f1 = 1/2 < b (N2) := by apply hN2.2; by  
[use N1 → N2  
intro n hn  
rewrite [hc]  
have a n \* b n > a n \* 1/2 := by bound

- Two midterms and one final exam, in class, timed, on paper!
- Mostly natural language, with a few Lean proofs thrown in.



# Real Analysis, The Game!

- I must reject the null hypothesis...

6. (10 pts) We say that a sequence  $a : \mathbb{N} \rightarrow \mathbb{R}$  diverges to infinity if it eventually exceeds any bound. More precisely:

```
def DivToInf (a :  $\mathbb{N} \rightarrow \mathbb{R}$ ) : Prop :=  $\forall M > 0, \exists N, \forall n \geq N, a n > M$ 
```

Give a formal proof of this fact, followed by a natural language statement (explaining what the theorem says) and natural language proof:

Objects:  
 $a \ b \ c : \mathbb{N} \rightarrow \mathbb{R}$   
 $L : \mathbb{R}$

Assumptions:  
(ha : DivToInf a)  $\forall M > 0, \exists N, \forall n \geq N, a n > M$   
(hb : SeqLim b L)  $\forall \epsilon > 0, \exists N, \forall n \geq N, |b n - L| < \epsilon$   
(hL :  $L > 0$ )  
(hc :  $\forall n, c n = a n * b n$ )  $hm : M > 0$

Goal: DivToInf c

intro M hm  
specialize hb (1/2) (by bound) ✓  
specialize ha (2M/L) (by bound) ✓  
choose N1 hN1 using ha  
choose N2 hN2 using hb  
specialize hN1 (N1) (by bound)  
specialize hN2 (N2) (by bound)  
rewrite [abs\_lt] at hN2  
have h1 : 1/2 < b (N2) := by apply hN2.2; by  
[use N1+N2  
intro n hn  
rewrite [hc]  
have a\_n \* b\_n > a\_n \* 1/2 := by bound

- Two midterms and one final exam, in class, timed, on paper!
- Mostly natural language, with a few Lean proofs thrown in.
- Towards the end, many *preferred* Lean proofs to the imprecision of natural language proofs! In the future: we will name all our *hypotheses*, and look back at math with unnamed hypotheses the same way we now look at 16th century algebra!

---

# Real Analysis, The Game!

- Towards the end, many preferred Lean proofs to the imprecision of natural language proofs! In the future: we will name all our *hypotheses*, and look back at math with unnamed hypotheses the same way we now look at 16th century algebra!
- Compare:
  - “You have some unknown quantity. When you create a second copy of this quantity and combine it with a known amount, then multiply this total by your original unknown quantity, you get a specified result”
- To:
  - $x(x+a)=b$ .



# Real Analysis, The Game!

- A peek into the future:...

The screenshot displays the Lean theorem prover interface, which includes a Scratch-like block editor and a code editor. The block editor on the left shows two theorems: 'away' and 'fun\_limit\_fact'. The 'away' theorem is defined as  $(y : \mathbb{R}) (hy : y \neq 1) : (y^2 - 1) / (y - 1) = y + 1$  and is proven using the 'grind' tactic. The 'fun\_limit\_fact' theorem is defined as  $\text{FunLimAt} (\text{fun } x \Rightarrow (x^2 - 1) / (x - 1)) 2 1$  and is proven using the 'Unfold', 'Intro', 'Use', 'pair', 'linarith', 'Intro', 'Intro', 'Intro', 'simp', and 'rewrite' tactics. The code editor on the right shows the corresponding Lean code for these theorems. The interface also includes a sidebar with 'Tactics', 'Variables', and 'Values' sections, and a bottom panel showing the 'Tactic state' and 'Messages'.

```
1 import Mathlib
2
3 def FunLimAt (f : ℝ → ℝ) (L : ℝ) (c : ℝ) : Prop :=
4   ∀ ε > 0, ∃ δ > 0, ∀ x ≠ c, |x - c| < δ → |f x - L| < ε
5
6 theorem away (y : ℝ) (hy : y ≠ 1) : (y^2 - 1) / (y - 1) = y + 1 := by
7   grind
8
9 theorem fun_limit_fact : FunLimAt (fun x => (x^2 - 1) / (x - 1)) 2 1 := by
10  unfold FunLimAt
11  intro ε
12  intro hc
```

- Real Analysis will be a Scratch-like game, with ~150 “escape room” puzzles that 12 year olds will compete to master... :)