The Prime Number Theorem

Manuel Eberl and Larry Paulson

September 13, 2023

Abstract

This article provides a short proof of the Prime Number Theorem in several equivalent forms, most notably $\pi(x) \sim x/\ln x$ where $\pi(x)$ is the number of primes no larger than $x$. It also defines other basic number-theoretic functions related to primes like Chebyshev’s $\vartheta$ and $\psi$ and the “$n$-th prime number” function $p_n$. We also show various bounds and relationship between these functions are shown. Lastly, we derive Mertens’ First and Second Theorem, i.e. $\sum_{p \leq x} \frac{\ln p}{p} = \ln x + O(1)$ and $\sum_{p \leq x} \frac{1}{p} = \ln \ln x + M + O(1/\ln x)$. We also give explicit bounds for the remainder terms.

The proof of the Prime Number Theorem builds on a library of Dirichlet series and analytic combinatorics. We essentially follow the presentation by Newman [6]. The core part of the proof is a Tauberian theorem for Dirichlet series, which is proven using complex analysis and then used to strengthen Mertens’ First Theorem to $\sum_{p \leq x} \frac{\ln p}{p} = \ln x + c + o(1)$.

A variant of this proof has been formalised before by Harrison in HOL Light [5], and formalisations of Selberg’s elementary proof exist both by Avigad et al. [2] in Isabelle and by Carneiro [3] in Metamath. The advantage of the analytic proof is that, while it requires more powerful mathematical tools, it is considerably shorter and clearer. This article attempts to provide a short and clear formalisation of all components of that proof using the full range of mathematical machinery available in Isabelle, staying as close as possible to Newman’s simple paper proof.
# Contents

1 Auxiliary material 3

2 Ingham’s Tauberian Theorem 37

3 Prime-Counting Functions 52
   3.1 Definitions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 53
   3.2 Basic properties . . . . . . . . . . . . . . . . . . . . . . . . . . 55
   3.3 The \( n \)-th prime number . . . . . . . . . . . . . . . . . 57
   3.4 Relations between different prime-counting functions . . . . 62
   3.5 Bounds . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 66
   3.6 Equivalence of various forms of the Prime Number Theorem . 74
   3.7 The asymptotic form of Mertens’ First Theorem . . . . . . . 79

4 The Prime Number Theorem 84
   4.1 Constructing Newman’s function . . . . . . . . . . . . . . . . . 84
   4.2 The asymptotic expansion of \( 2 \) . . . . . . . . . . . . . . . . 93
   4.3 The asymptotics of the prime-counting functions . . . . . . . 99

5 Mertens’ Theorems 101
   5.1 Absolute Bounds for Mertens’ First Theorem . . . . . . . . . . 102
   5.2 Mertens’ Second Theorem . . . . . . . . . . . . . . . . . . . . . 113

6 Acknowledgements 118
1 Auxiliary material

theory Prime-Number-Theorem-Library
imports
  Zeta-Function.Zeta-Function
  HOL−Real-Asymp.Real-Asymp
begin

Conflicting notation from HOL−Analysis.Infinite-Sum

no-notation Infinite-Sum.abs-summable-on (infix abs'-summable'-on 46)

lemma homotopic-loopsI:
fixes h :: real × real
assumes continuous-on (\{0..1\} × \{0..1\}) h
\forall x. x ∈ \{0..1\} ⇒ h (0, x) = p x
\forall x. x ∈ \{0..1\} ⇒ h (1, x) = q x
\forall x. x ∈ \{0..1\} ⇒ pathfinish (h o Pair x) = pathstart (h o Pair x)
shows homotopic-loops s p q
using assms unfolding homotopic-loops by (intro exI[of - h]) auto

lemma homotopic-pathsI:
fixes h :: real × real
assumes continuous-on (\{0..1\} × \{0..1\}) h
assumes h ' (\{0..1\} × \{0..1\}) ⊆ s
assumes \forall x. x ∈ \{0..1\} ⇒ h (0, x) = p x
assumes \forall x. x ∈ \{0..1\} ⇒ h (1, x) = q x
assumes \forall x. x ∈ \{0..1\} ⇒ pathstart (h o Pair x) = pathstart p
assumes \forall x. x ∈ \{0..1\} ⇒ pathfinish (h o Pair x) = pathfinish p
shows homotopic-paths s p q
using assms unfolding homotopic-paths by (intro exI[of - h]) auto

lemma sum-upto-ln-conv-sum-upto-mangoldt:
sum-upto (λn. ln (real n)) x = sum-upto (λn. mangoldt n * nat ⌊x / real n⌋) x
proof −
  have sum-upto (λn. ln (real n)) x =
    sum-upto (λn. \sum d \mid d dvd n. mangoldt d) x
    by (intro sum-upto-cong) (simp-all add: mangoldt-sum)
  also have \dots = sum-upto (λk. sum-upto (λd. mangoldt k) (x / real k)) x
    by (rule sum-upto-sum-divisors)
  also have \dots = sum-upto (λn. mangoldt n * nat ⌊x / real n⌋) x
    unfolding sum-upto-altdef by (simp add: mult-ac)
finally show ?thesis.
qed

lemma ln-fact-conv-sum-upto-mangoldt:
ln (fact n) = sum-upto (λk. mangoldt k * (n div k)) n
proof −
  have [simp]: \{0<..<Suc n\} = insert (Suc n) \{0<..<n\} for n by auto
have \( \ln (\text{fact } n) = \sumupto (\lambda n. \ln (\text{real } n)) n \)
by (induction \( n \)) (auto simp: \text{sumupto-altdef } \text{nat-add-distrib } \text{ln-mult})
also have \( \ldots = \sumupto (\lambda k. \text{mangoldt } k \times (n \div k)) n \)
unfolding \( \sumupto-ln-conv-sumupto-mangoldt \)
by (intro \text{sumupto-cong}) (auto simp: \text{floor-divide-of-nat-eq})
finally show \( \text{thesis} \).

qed

lemma \( \text{fds-abs-converges-comparison-test} \):
  fixes \( s \) :: 'a :: dirichlet-series
  assumes eventually (\( \lambda n. \text{norm } (\text{fds-nth } f n) \leq \text{fds-nth } g n \) at-top \) \( \text{and } \text{fds-converges } g (s \cdot 1) \)
  shows \( \text{fds-abs-converges } f s \)
unfoldling \( \text{fds-abs-converges-def} \)
proof (rule \text{summable-comparison-test-ev})
  from \text{assms}(2) \text{ show summable } (\lambda n. \text{fds-nth } g n / n \text{ powr } (s \cdot 1))
  by (auto simp: \text{fds-converges-def})
  from \text{assms}(1) \text{ eventually-gt-at-top}[of 0]
  show eventually \( (\lambda n. \text{norm } (\text{norm } (\text{fds-nth } f n / \text{nat-power } n s)) \leq \text{fds-nth } g n / \text{real } n \text{ powr } (s \cdot 1)) \) at-top
  by eventually-elim (auto simp: \text{norm-divide} \text{ norm-nat-power} intro: \text{divide-right-mono})

qed

lemma \( \text{fds-converges-scaleR} \) [intro]:
  assumes \( \text{fds-converges } f s \)
  shows \( \text{fds-converges } (c *R f) s \)
proof –
  from \text{assms} have \text{summable } (\lambda n. c *R (\text{fds-nth } f n / \text{nat-power } n s))
  by (intro \text{summable-scaleR-right}) (auto simp: \text{fds-converges-def})
  also have \( (\lambda n. c *R (\text{fds-nth } f n / \text{nat-power } n s)) = (\lambda n. (c *R \text{fds-nth } f n / \text{nat-power } n s)) \)
  by (simp add: \text{scaleR-conv-of-real})
  finally show \( \text{thesis} \) by (simp add: \text{fds-converges-def})

qed

lemma \( \text{fds-abs-converges-scaleR} \) [intro]:
  assumes \( \text{fds-abs-converges } f s \)
  shows \( \text{fds-abs-converges } (c *R f) s \)
proof –
  from \text{assms} have \text{summable } (\lambda n. \text{abs } c * \text{norm } (\text{fds-nth } f n / \text{nat-power } n s))
  by (intro \text{summable-mult}) (auto simp: \text{fds-abs-converges-def})
  also have \( (\lambda n. \text{abs } c * \text{norm } (\text{fds-nth } f n / \text{nat-power } n s)) = (\lambda n. \text{norm } ((c *R \text{fds-nth } f n) / \text{nat-power } n s)) \)
  by (simp add: \text{norm-divide})
  finally show \( \text{thesis} \) by (simp add: \text{fds-abs-converges-def})

qed

lemma \( \text{conv-abscissa-scaleR} \): \( \text{conv-abscissa } (\text{scaleR } c f) \leq \text{conv-abscissa } f \)
by (rule \text{conv-abscissa-mono}) auto
lemma \textit{abs-conv-abscissa-scaleR}: 
\textit{abs-conv-abscissa} (scaleR \ c \ f) \leq \textit{abs-conv-abscissa} \ f

by (rule \textit{abs-conv-abscissa-mono}) auto

lemma \textit{fds-conv-converges-mult-const-left} [intro]:
fds-conv-converges \ f \ s \Longrightarrow \textit{fds-conv-converges} (fds-const \ c \ * \ f) \ s

by (auto simp: \textit{fds-conv-converges-def} norm-mult norm-divide dest: summable-mult[of - norm \ c])

lemma \textit{conv-abscissa-mul-const-left}:
\textit{conv-abscissa} (fds-const \ c \ * \ f) \leq \textit{conv-abscissa} \ f

by (intro \textit{conv-abscissa-mono}) auto

lemma \textit{bounded-coeffs-imp-fds-conv-converges}:

\textit{bounded-coeffs-imp-fds-conv-converges}

fixes \ s :: \ 'a :: dirichlet-series and \ f :: 'a fds

assumes \texttt{Bseq} (fds-nth \ f) \ s \cdot 1 > 1

shows \ fds-conv-converges \ f \ s

proof
  from \textit{asms} obtain \ C \ where \ C: \ \forall n. \ norm (fds-nth \ f \ n) \leq \ C
  by (auto simp: \textit{Bseq-def})

  show \ ?thesis
  proof (rule \textit{fds-conv-converges-comparison-test})
    from \(s \cdot 1 > 1\) show \textit{fds-conv-converges} (C *R fds-zeta) (s \cdot 1)
    by (intro \textit{fds-conv-converges-imp-converges}) auto

    from \ C \ show \ eventually \ (\lambda n. \ norm (fds-nth \ f \ n) \leq \textit{fds-nth} \ (C *R \textit{fds-zeta}) \ n)
    at-top
      by (intro always-eventually) (auto simp: \textit{fds-nth-zeta})

  qed

  qed

lemma \textit{bounded-coeffs-imp-fds-conv-converges}':
fixes $s :: 'a :: dirichlet-series$ and $f :: 'a fds$
assumes $Bseq \{\lambda n. fds-nth f n * nat-power n s0\} s \cdot 1 > 1 - s0 \cdot 1$
shows $fds-abs-converges f s$

proof |
  have $(fds-nth \ (fds-shift s0 f)) = (\lambda n. fds-nth f n * nat-power n s0)$
    by (auto simp: fan-eq-iff)
  with assms have $Bseq \ (fds-nth \ (fds-shift s0 f))$ by simp
  with assms(2) have $fds-abs-converges \ (fds-shift s0 f) \ (s + s0)$
    by (intro bounded-coeffs-imp-fds-abs-converges) (auto simp: algebra-simps)
thus $\text{thesis}$ by simp
qed

lemma bounded-coeffs-imp-abs-conv-abscissa-le:
  fixes $s :: 'a :: dirichlet-series$ and $f :: 'a fds$ and $c :: \text{ereal}$
assumes $Bseq \ (\lambda n. fds-nth f n)$
shows $\text{abs-conv-abscissa f} \leq c$

proof |
  fix $x$ assume $c < \text{ereal x}$
  have $\text{ereal} \ (1 - s \cdot 1) \leq c$ by fact
  also have $\ldots < \text{ereal x}$ by fact
  finally have $1 - s \cdot 1 < \text{ereal x}$ by simp
  thus $fds-abs-converges f \ (\text{of-real x})$
    by (intro bounded-coeffs-imp-fds-abs-converges)[OF assms(1)] auto
qed

lemma bounded-coeffs-imp-abs-conv-abscissa-le-1:
  fixes $s :: 'a :: dirichlet-series$ and $f :: 'a fds$
assumes $Bseq \ (\lambda n. fds-nth f n)$
shows $\text{abs-conv-abscissa f} \leq 1$

proof |
  have [simp]: $fds-nth f n * nat-power n 0 = fds-nth f n$ for $n$
    by (cases $n = 0$) auto
  show $\text{thesis}$
    by (rule bounded-coeffs-imp-abs-conv-abscissa-le[where $s = 0$]) (insert assms, auto simp:)
qed

lemma
defines $a$ $b$ $c :: \text{real}$
assumes $ab: a + b > 0$ and $c :: -1$
shows set-integrable-powr-at-top: $(\lambda x. b + x) \ powr c$ absolutely-integrable-on
    {a<..<}
and set-lebesgue-integrable-powr-at-top:
    $(\int x\in\{a<..<\}. \ ((b + x) powr c) \ \partial lborel) = -((b + a) powr (c + 1) / (c + 1))$
and powr-has-integral-at-top:
    $(\lambda x. b + x) powr c$ has-integral $-((b + a) powr (c + 1) / (c + 1))$
    {a<..<}
proof –
  
  let \( \mathcal{F} = \lambda x. (b + x) \) and \( \mathcal{F} = \lambda x. (b + x) \) powr \((c + 1) / (c + 1)\)

  have limits: ((\( \mathcal{F} \circ \text{real-of-ereal} \)) \( \rightarrow \mathcal{F} \) a) (at-right (ereal a))

  using c ab unfolding ereal-tendsto-simps1 by (real-asymp simp: field-simps)+

  have 1: set-integrable borel (einterval a \( \infty \)) \( \mathcal{F} \) by (intro interval-integral-FTC-nonneg) (auto intro: derivative-eq-intros)

  thus 2: \( \mathcal{F} \) absolutely-integrable-on \( \{a<..\} \)

  by (auto simp: interval-integral-to-infinity-eq)

  show \( \mathcal{F} \) has-integral \( -(((b + a) \) powr (c + 1) / (c + 1)) \)

  by (simp add: has-integral-iff)

  qed

def lemma fds-converges-altdef2:
  fds-converges f s \( \leftrightarrow \) convergent \( \lambda N. \) eval-fds (fds-truncate N f) s

  unfolding fds-converges-def summable-iff-convergent' eval-fds-truncate

  by (auto simp: not-le intro: convergent-cong always-eventually sum mono-neutral-right)

lemma tendsto-eval-fds-truncate:
  assumes fds-converges f s

  shows \( \lambda N. \) eval-fds (fds-truncate N f) s \( \longrightarrow \) eval-fds f s

  proof –

  have \( \lambda N. \) eval-fds (fds-truncate N f) s \( \longrightarrow \) eval-fds f s

  unfolding eval-fds-truncate

  by (intro filterlim-cong always-eventually allI sum mono-neutral-left) (auto simp: not-le)

  also have . . using assms

  by (simp add: fds-converges-iff sums-def' atLeast0AtMost)

  finally show \( ?\)thesis .

  qed

lemma linepath-translate-left: linepath \( (c + a) \) (c + a) = \( \lambda x. c + a \) o linepath a b

  by auto

lemma linepath-translate-right: linepath \( (a + c) \) (b + c) = \( \lambda x. x + c \) o linepath a b

  by (auto simp: fun-eq-iff linepath-def algebra-simps)

lemma has-contour-integral-linepath-same-Im-iff:
  fixes a b :: complex and f :: complex \( \Rightarrow \) complex

  assumes Im a = Im b Re a < Re b

  shows \( f \) has-contour-integral \( I \) (linepath a b) \( \leftrightarrow \)
\((\lambda x. f (\text{of-real } x + \text{Im } a \ast i))\) has-integral \(I\) \{\text{Re } a..\text{Re } b\}

**proof**

- have deriv: vector-derivative \((\lambda x. x - \text{Im } a \ast i) \circ \text{linepath } a \text{ b}\) (at \(y\)) = \(b - a\) for \(y\)
  - using linepath-translate-right[of \(a - \text{Im } a \ast i\) \(b\), symmetric] by simp
  - have \((f\text{ has-contour-integral }I)\) (linepath \(a \text{ b}\) \(\iff\) \((\lambda x. f (x + \text{Im } a \ast i))\text{ has-contour-integral }I\) (linepath \(a - \text{Im } a \ast i\) \(b - \text{Im } a \ast i\))
    - using linepath-translate-right[of \(a - \text{Im } a \ast i\) \(b\)] deriv by (simp add: has-contour-integral)
  - also have \(\ldots \iff (\lambda x. f (x + \text{Im } a \ast i))\text{ has-integral }I\) \{\text{Re } a..\text{Re } b\} using assms
    - by (subst has-contour-integral-linepath-Reals-iff) (auto simp: complex-is-Real-iff)
  - finally show \(?\text{thesis}\).

**qed**

**lemma** contour-integrable-linepath-same-Im-iff:

- fixes \(a \text{ b : complex and } f :: \text{complex \Rightarrow complex}\)
- assumes \(\text{Im } a = \text{Im } b \text{ Re } a < \text{Re } b\)
- shows \((f\text{ contour-integrable-on linepath } a \text{ b}) \iff (\lambda x. f (\text{of-real } x + \text{Im } a \ast i))\text{ integrable-on }\{\text{Re } a..\text{Re } b\}\)
  - using contour-integrable-on-def has-contour-integral-linepath-same-Im-iff[OF assms] by blast

**lemma** contour-integral-linepath-same-Im:

- fixes \(a \text{ b : complex and } f :: \text{complex \Rightarrow complex}\)
- assumes \(\text{Im } a = \text{Im } b \text{ Re } a < \text{Re } b\)
- shows contour-integral (linepath \(a \text{ b}\) \(f\) = integral \{\text{Re } a..\text{Re } b\} (\lambda x. f (\text{of-real } x + \text{Im } a \ast i))\)
  - proof (cases \(f\text{ contour-integrable-on linepath } a \text{ b}\))
    - case True
      - thus \(?\text{thesis}\) using has-contour-integral-linepath-same-Im-iff[OF assms, of \(f\)] using has-contour-integral-integral has-contour-integral-unique by blast
    - next
      - case False
        - thus \(?\text{thesis}\) using contour-integrable-linepath-same-Im-iff[OF assms, of \(f\)] by (simp add: not-integrable-contour-integral not-integrable-integral)

**qed**

**lemmas** [simp del] = div-mult-self3 div-mult-self4 div-mult-self2 div-mult-self1

**interpretation** cis: periodic-fun-simple cis 2 * pi

- by standard (simp-all add: complex-eq-iff)

**lemma** analytic-onE-box:

- assumes \(f\text{ analytic-on } A\) \(s \in A\)
- obtains \(a \text{ b where } \text{Re } a < \text{Re } b \text{ Im } a < \text{Im } b\) \(s \in \text{box } a \text{ b}\) \(f\text{ analytic-on box } a \text{ b}\)
  - proof
    - from assms obtain \(r\) where \(r > 0\) holomorphic-on ball \(s\) \(r\)

8
by (auto simp: analytic-on-def)
with open-contains-box[of ball s r s] obtain a b
  where box a b ⊆ ball s r s ∈ box a b ∀ i ∈ Basis. a · i < b · i by auto
moreover from r have f analytic-on ball s r by (simp add: analytic-on-open)
ultimately show ?thesis using that[of a b] analytic-on-subset[of - ball s r box a b]
  by (auto simp: Basis-complex-def)
qed

lemma Re-image-box:
  assumes Re a < Re b Im a < Im b
  shows Re ' box a b = {Re a <..< Re b}
  using inner-image-box[of 1::complex a b] assms by (auto simp: Basis-complex-def)

lemma Im-image-box:
  assumes Re a < Re b Im a < Im b
  shows Im ' box a b = {Im a <..< Im b}
  using inner-image-box[of i::complex a b] assms by (auto simp: Basis-complex-def)

lemma Re-image-cbox:
  assumes Re a ≤ Re b Im a ≤ Im b
  shows Re ' cbox a b = {Re a .. Re b}
  using inner-image-cbox[of 1::complex a b] assms by (auto simp: Basis-complex-def)

lemma Im-image-cbox:
  assumes Re a ≤ Re b Im a ≤ Im b
  shows Im ' cbox a b = {Im a .. Im b}
  using inner-image-cbox[of i::complex a b] assms by (auto simp: Basis-complex-def)

lemma analytic-onE-cball:
  assumes f analytic-on A s ∈ A ub > (0::real)
  obtains R where R > 0 R < ub f analytic-on cball s R
  proof –
    from assms obtain r where r > 0 f holomorphic-on ball s r
      by (auto simp: analytic-on-def)
    hence f analytic-on ball s r by (simp add: analytic-on-open)
    hence f analytic-on cball s (min (ub / 2) (r / 2))
      by (rule analytic-on-subset, subst cball-subset-ball-iff) (use r > 0 in auto)
    moreover have min (ub / 2) (r / 2) > 0 and min (ub / 2) (r / 2) < ub
      using r > 0 and ub > 0 by (auto simp: min-def)
    ultimately show ?thesis using that[of min (ub / 2) (r / 2)]
      by blast
  qed

corollary analytic-pre-zeta' [analytic-intros]:
  assumes f analytic-on A a > 0
  shows (λx. pre-zeta a (f x)) analytic-on A
  using analytic-on-compose-gen[OF assms(1) analytic-pre-zeta[of a UNIV]] assms(2)
by (auto simp: o-def)

corollary analytic-hurwitz-zeta' [analytic-intros]:
  assumes f analytic-on A (\(\forall x. x \in A \implies f x \neq 1\)) \(a > 0\)
  shows (\(\lambda x. \text{hurwitz-zeta} a (f x)\)) analytic-on A
  using analytic-on-compose-gen[OF assms(1) analytic-hurwitz-zeta[of a \{-1\}]]
  assms(2,3)
  by (auto simp: o-def)

corollary analytic-zeta' [analytic-intros]:
  assumes f analytic-on A (\(\forall x. x \in A \implies f x \neq 1\))
  shows (\(\lambda x. \text{zeta} (f x)\)) analytic-on A
  using analytic-on-compose-gen[OF assms(1) analytic-zeta[of \{-1\}]] assms(2)
  by (auto simp: o-def)

lemma logderiv-zeta-analytic: (\(\lambda s. \text{deriv} \ zeta s / \zeta s\)) analytic-on \(\{s. \Re s \geq 1\} - \{1\}\)
  using zeta-Re-ge-1-nonzero by (auto intro: analytic-intros)

lemma mult-real-sqrt: \(x \geq 0 \implies x \ast \sqrt y = \sqrt (x ^ 2 \ast y)\)
  by (simp add: real-sqrt-mult)

lemma arcsin-pos: \(x \in \{0 <..1\} \implies \arcsin x > 0\)
  using arcsin-less-arcsin[of 0 x] by simp

lemmas analytic-imp-holomorphic' = holomorphic-on-subset[OF analytic-imp-holomorphic]

lemma residue-simple':
  assumes open s 0 \(\in\) s f holomorphic-on s
  shows residue (\(\lambda w. f w / w\)) 0 = f 0
  using residue-simple[of s 0 f] assms by simp

lemma fds-converges-cong:
  assumes eventually (\(\lambda n. \text{fds-nth} f n = \text{fds-nth} g n\)) at-top s = s'
  shows \(\lambda s. \text{fds-converges} f s \longleftrightarrow \text{fds-converges} g s'\)
  unfolding fds-converges-def
  by (intro summable-cong eventually-mono[OF assms(1)]) (simp-all add: assms)

lemma fds-abs-converges-cong:
  assumes eventually (\(\lambda n. \text{fds-nth} f n = \text{fds-nth} g n\)) at-top s = s'
  shows \(\lambda s. \text{fds-abs-converges} f s \longleftrightarrow \text{fds-abs-converges} g s'\)
  unfolding fds-abs-converges-def
  by (intro summable-cong eventually-mono[OF assms(1)]) (simp-all add: assms)

lemma conv-abscissa-cong:
  assumes eventually (\(\lambda n. \text{fds-nth} f n = \text{fds-nth} g n\)) at-top
  shows \(\lambda s. \text{conv-abscissa} f = \text{conv-abscissa} g\)
proof
  have fds-converges f = fds-converges g
    by (intro ext fds-converges-cong assms refl)
  thus ?thesis by (simp add: conv-abscissa-def)
qed

lemma abs-conv-abscissa-cong:
  assumes eventually (λn. fds-nth f n = fds-nth g n) at-top
  shows abs-conv-abscissa f = abs-conv-abscissa g
proof
  have fds-abs-converges f = fds-abs-converges g
    by (intro ext fds-abs-converges-cong assms refl)
  thus ?thesis by (simp add: abs-conv-abscissa-def)
qed

definition fds-remainder where
  fds-remainder m = fds-subseries (λn. n > m)

lemma fds-nth-remainder: fds-nth (fds-remainder m f) = (λn. if n > m then fds-nth f n else 0)
  by (simp add: fds-remainder-def fds-subseries-def fds-nth-fds')

lemma fds-converges-remainder-iff [simp]:
  fds-converges (fds-remainder m f) s iff fds-converges f s
  by (intro fds-converges-cong eventually-mono[OF eventually-gt-at-top[of m]])
    (auto simp: fds-nth-remainder)

lemma fds-abs-converges-remainder-iff [simp]:
  fds-abs-converges (fds-remainder m f) s iff fds-abs-converges f s
  by (intro fds-abs-converges-cong eventually-mono[OF eventually-gt-at-top[of m]])
    (auto simp: fds-nth-remainder)

lemma fds-converges-remainder [intro]:
  fds-converges f s implies fds-converges (fds-remainder m f) s
and fds-abs-converges-remainder [intro]:
  fds-abs-converges f s implies fds-abs-converges (fds-remainder m f) s
  by simp-all

lemma conv-abscissa-remainder [simp]:
  conv-abscissa (fds-remainder m f) = conv-abscissa f
  by (intro conv-abscissa-cong eventually-mono[OF eventually-gt-at-top[of m]])
    (auto simp: fds-nth-remainder)

lemma abs-conv-abscissa-remainder [simp]:
  abs-conv-abscissa (fds-remainder m f) = abs-conv-abscissa f
  by (intro abs-conv-abscissa-cong eventually-mono[OF eventually-gt-at-top[of m]])
    (auto simp: fds-nth-remainder)
lemma eval-fds-remainder:
  eval-fds (fds-remainder m f) s = (∑ n. fds-nth f (n + Suc m) / nat-power (n + Suc m) s)
  (is s = suminf (λn. ?f (n + Suc m)))
proof (cases fds-converges f s)
case False
  hence ¬fds-converges (fds-remainder m f) s by simp
  hence (λx. (λn. fds-nth (fds-remainder m f) n / nat-power n s) sums x) = (λx. False) by (auto simp: fds-converges-def summable-def)
moreover from False have ¬summable (λn. ?f (n + Suc m)) unfolding fds-converges-def
by (subst suminf-minus-initial-segment) (auto simp: fds-nth-remainder)
also have (λn. fds-nth (fds-remainder m f) (n + Suc m)) = (λn. fds-nth f (n + Suc m)) by (intro ext) (auto simp: fds-nth-remainder)
ultimately show ?thesis by simp
next
case True
  hence *: fds-converges (fds-remainder m f) s by simp
  have eval-fds (fds-remainder m f) s = (∑ n. fds-nth (fds-remainder m f) n / nat-power n s)
  unfolding eval-fds-def ..
  also have ... = (∑ n. fds-nth (fds-remainder m f) (n + Suc m) / nat-power (n + Suc m) s)
  using * unfolding fds-converges-def
  by (auto simp: fds-nth-remainder)
  also have (λn. fds-nth (fds-remainder m f) (n + Suc m)) = (λn. fds-nth f (n + Suc m))
  by (intro ext) (auto simp: fds-nth-remainder)
  finally show ?thesis .

qed

lemma fds-truncate-plus-remainder: fds-truncate m f + fds-remainder m f = f
by (intro fds-eqI) (auto simp: fds-truncate-def fds-remainder-def fds-subseries-def)

lemma holomorphic-fds-eval’ [holomorphic-intros]:
assumes g holomorphic-on A ∧ x. x ∈ A ⊢ Re (g x) > conv-abscissa f
shows (λx. eval-fds f (g x)) holomorphic-on A
using holomorphic-on-compose-gen[OF assms(1)] holomorphic-fds-eval[OF order.refl, of f] assms(2)
by (auto simp: o-def)

lemma analytic-fds-eval’ [analytic-intros]:
assumes g analytic-on A ∧ x. x ∈ A ⊢ Re (g x) > conv-abscissa f
shows \((\lambda x. \text{eval-fds } f \ (g \ x))\) analytic-on \(A\)
using analytic-on-compose-gen[OF assms(1) analytic-fds-eval[OF order.refl, of \(f\)]
by (auto simp: o-def)

lemma continuous-on-linepath [continuous-intros]:
assumes continuous-on \(A\) \(a\) continuous-on \(A\) \(b\) continuous-on \(A\) \(f\)
shows continuous-on \(A\) \((\lambda x. \text{linepath } (a \ x) \ (b \ x) \ (f \ x))\)
using assms by (auto simp: linepath-def intro: continuous-intros assms)

lemma continuous-on-part-circlepath [continuous-intros]:
assumes continuous-on \(A\) \(c\) continuous-on \(A\) \(r\) continuous-on \(A\) \(a\) continuous-on \(A\) \(b\) continuous-on \(A\) \(f\)
shows continuous-on \(A\) \((\lambda x. \text{part-circlepath } c \ r \ a \ x \ b \ x \ (f \ x))\)
using assms by (auto simp: part-circlepath-def intro: continuous-intros assms)

lemma homotopic-loops-part-circlepath:
assumes sphere \(c\) \(r\) \(\subseteq\) \(A\) and \(r\) \(\geq\) \(0\) and \(\text{b1} = a1 + 2 \ast \text{of-int } k \ast \pi\) and \(\text{b2} = a2 + 2 \ast \text{of-int } k \ast \pi\)
shows homotopic-loops \(A\) \((\text{part-circlepath } c \ r \ a1 \ b1) \ (\text{part-circlepath } c \ r \ a2 \ b2)\)
proof –
define \(h\) where \(h = (\lambda(x,y). \text{part-circlepath } c \ r \ (\text{linepath } a1 \ a2 \ x) \ (\text{linepath } b1 \ b2 \ x) \ y)\)
show \(?thesis\)
proof (rule homotopic-loopsI)
show continuous-on \((\{0\..1\} \times \{0\..1\}) \ h\)
by (auto simp: \(h\)-def case-prod-unfold intro: continuous-intros)
next
from assms have \(h \ ((\{0\..1\} \times \{0\..1\}) \subseteq \text{sphere } c \ r\)
by (auto simp: \(h\)-def part-circlepath-def dist-norm norm-mult)
also have \(\ldots \subseteq \text{A by fact}\)
finally show \(h \ ((\{0\..1\} \times \{0\..1\}) \subseteq \text{A .}\)
next
fix \(x::\) real assume \(x. \ x \in \{0\..1\}\)
show \(h \ (0, x) = \text{part-circlepath } c \ r \ a1 \ b1 \ x \ \text{and} \ h \ (1, x) = \text{part-circlepath } c \ r \ a2 \ b2 \ x\)
by (simp-all add: \(h\)-def linepath-def
have \(\text{cis } (\pi \ast \text{of-int } k \ast 2)) = 1\)
using \(\text{cis.plus-of-int}\{0 \ k\}\) by (simp add: algebra-simps)
thus pathfinish \((h \circ \text{Pair } \ x) = \text{pathstart} \ (h \circ \text{Pair } \ x)\)
by (simp add: \(h\)-def o-def exp-eq-polar linepath-def algebra-simps
  \text{cis-mult } [\text{symmetric} \text{cis-divide } [\text{symmetric} \text{assms}])
qed

lemma part-circlepath-cone-subpath:
part-circlepath \(c \ r \ a \ b = \text{subpath } (a / (2\ast\pi)) \ (b / (2\ast\pi))\) \(\text{circlepath } c \ r\)
by (simp add: part-circlepath-def circlepath-def subpath-def linepath-def alge-
lemma homotopic-paths-part-circlepath:
assumes $a \leq b \leq c$
assumes path-image $(\text{part-circlepath} C r a c) \subseteq A$ $r \geq 0$
shows homotopic-paths $A$ $(\text{part-circlepath} C r a c)$
$(\text{part-circlepath} C r a b +++ \text{part-circlepath} C r b c)$
(is homotopic-paths $?g$ $(?h1 +++ ?h2)$)
proof (cases $a = c$)
case False
with assms have $a < c$ by simp
define slope where $\text{slope} = (b - a) / (c - a)$
from assms and $\langle a < c \rangle$ have slope: $\text{slope} \in \{0..1\}$
  by (auto simp: field-simps slope-def)
define $f :: \text{real} \Rightarrow \text{real}$ where $f = \text{linepath} 0 \text{slope} +++ \text{linepath} \text{slope} 1$
show ?thesis
proof (rule homotopic-paths-reparametrize)
  fix $t :: \text{real}$ assume $t \in \{0..1\}$
  show $(?h1 +++ ?h2) t = ?g (f t)$
  proof (cases $t \leq 1 / 2$)
    case True
    hence $?g (f t) = C + r * \text{cis} ((1 - f t) * a + f t * c)$
      by (simp add: joinpaths-def part-circlepath-def exp-eq-polar linepath-def)
    also from True $\langle a < c \rangle$ have $(1 - f t) * a + f t * c = (1 - 2 * t) * a + 2 * t * b$
      unfolding $f$-def slope-def linepath-def joinpaths-def
      (simp add: algebra-simps)\
    also from True have $C + r * \text{cis} \ldots = (?h1 +++ ?h2) t$
      by (simp add: joinpaths-def part-circlepath-def exp-eq-polar linepath-def)
    finally show ?thesis ..
  next
    case False
    hence $?g (f t) = C + r * \text{cis} ((1 - f t) * a + f t * c)$
      by (simp add: joinpaths-def part-circlepath-def exp-eq-polar linepath-def)
    also from False $\langle a < c \rangle$ have $(1 - f t) * a + f t * c = (2 - 2 * t) * b + (2 * t - 1) * c$
      unfolding $f$-def slope-def linepath-def joinpaths-def
      (simp add: algebra-simps)\
    also from False have $C + r * \text{cis} \ldots = (?h1 +++ ?h2) t$
      by (simp add: joinpaths-def part-circlepath-def exp-eq-polar linepath-def)
    finally show ?thesis ..
  qed
next

14
from slope have path-image \( f \subseteq \{0..1\} \)
    by (auto simp: f-def path-image-join closed-segment-eq-real-ivl)
thus \( f \in \{0..1\} \rightarrow \{0..1\} \) by (force simp add: path-image-def)
next
  have path f unfolding f-def by auto
  thus continuous-on \( \{0..1\} \) f by (simp add: path-def)
qed (insert assms, auto simp: f-def joinpaths-def linepath-def)

next
  case [simp]: True
  with assms have [simp]: \( b = c \) by auto
  have part-circlepath \( C \) r c c +++ part-circlepath \( C \) r c c = part-circlepath \( C \) r c c
    by (simp add: fun-eq-iff joinpaths-def part-circlepath-def)
  thus \(?thesis\) using assms by simp
qed

lemma path-image-part-circlepath-subset:
  assumes \( a \leq a' \) \( a' \leq b' \) \( b' \leq b \)
  shows path-image (part-circlepath c r a' b') \( \subseteq \) path-image (part-circlepath c r a b)
  using assms by (subst (1 2) path-image-part-circlepath) auto

lemma part-circlepath-mirror:
  assumes \( a' = a + \pi \) \( a' \leq b' \) \( b' \leq b \)
  shows \( -\) part-circlepath c r a b = part-circlepath c' r a' b'
proof
  fix \( x :: \text{real} \)
  have part-circlepath \( c' \) r a' b' x = \( c' + r * \text{cis} (\text{linepath a b x} + \pi) + k \) \* \( 2 * \pi \)
    by (simp add: part-circlepath-def assms exp-eq-polar linepath-translate-right mult-ac)
  also have \( \text{cis} (\text{linepath a b x} + \pi) + k \) \* \( 2 * \pi \) = \( \text{cis} (\text{linepath a b x} + \pi) \)
    by (rule cis.plus-of-int)
  also have \( \ldots = -\text{cis} (\text{linepath a b x}) \)
    by (simp add: minus-cis)
  also have \( c' + r \ldots = -\text{part-circlepath c r a b} \)
    by (simp add: part-circlepath-mirror assms exp-eq-polar)
  finally show \( -\text{part-circlepath c r a b} \) \( x = \text{part-circlepath c' r a' b'} \)
    by simp
qed

lemma path-mirror [intro]: path \((g :: \Rightarrow \text{topological-group-add}) \Rightarrow \text{path} (-g)\)
  by (auto simp: path-def intro: continuous-intros)

lemma path-mirror-iff [simp]: path \((-g :: \Rightarrow \text{topological-group-add}) \leftrightarrow \text{path} \ g\)
  using path-mirror[of g] path-mirror[of -g] by (auto simp: fun-Compl-def)
lemma valid-path-mirror [intro]: valid-path $g$ $\Rightarrow$ valid-path $(-g)$ 
by (auto simp: valid-path-def fun-Compl-def piecewise-C1-differentiable-neg)

lemma valid-path-mirror-iff [simp]: valid-path $(-g)$ $\iff$ valid-path $g$
using valid-path-mirror [of $g$] valid-path-mirror [of $-g$] by (auto simp: fun-Compl-def)

lemma pathstart-mirror [simp]: pathstart $(-g)$ = $-pathstart g$
and pathfinish-mirror [simp]: pathfinish $(-g)$ = $-pathfinish g$
by (simp-all add: pathstart-def pathfinish-def)

lemma path-image-mirror: path-image $(-g)$ = uminus ' path-image $g$
by (auto simp: path-image-def)

lemma cos-le-zero: assumes $x \in \{\pi/2..3*\pi/2\}$ shows $\cos x \leq 0$
proof -
  have $\cos x = -\cos (x - \pi)$ by (simp add: cos-diff)
  moreover from assms have $\cos (x - \pi) \geq 0$
    by (intro cos-ge-zero) auto
  ultimately show $\?thesis$ by simp
qed

lemma cos-le-zero': $x \in \{-3*\pi/2..-\pi/2\}$ $\Rightarrow$ $\cos x \leq 0$
using cos-le-zero [of $-x$] by simp

lemma winding-number-join-pos-combined':
  \[\begin{align*}
  &\text{valid-path } \gamma_1 \land z \notin \text{path-image } \gamma_1 \land 0 < \text{Re}(\text{winding-number } \gamma_1 z); \\
  &\text{valid-path } \gamma_2 \land z \notin \text{path-image } \gamma_2 \land 0 < \text{Re}(\text{winding-number } \gamma_2 z); \\
  &\text{pathfinish } \gamma_1 = \text{pathstart } \gamma_2 \\
  \Rightarrow &\text{valid-path}(\gamma_1 +++ \gamma_2) \land z \notin \text{path-image}(\gamma_1 +++ \gamma_2) \land 0 < \text{Re}(\text{winding-number}(\gamma_1 +++ \gamma_2) z) \\
  \end{align*}\]
by (simp add: valid-path-join path-image-join winding-number-join valid-path-imp-path)

lemma Union-atLeastAtMost-real-of-nat: assumes $a < b$
shows $\bigcup n \in \{a..b\}. \{\text{real } n..\text{real } (n + 1)\} = \{\text{real } a..\text{real } b\}$
proof (intro equalityI subsetI)
  fix $x$ assume $x \in \{\text{real } a..\text{real } b\}$
  thus $x \in \bigcup n \in \{a..b\}. \{\text{real } n..\text{real } (n + 1)\}$
proof (cases $x = \text{real } b$
  case True
    with assms show $\?thesis$ by (auto intro!: bexI[of - $b - 1$])
next
  case False
  with $x$ have $x : x \geq \text{real } a < \text{real } b$ by simp-all
  hence $x \geq \text{real } (\text{nat } [x]) \leq \text{real } (\text{Suc } (\text{nat } [x])$ by linarith+
  moreover from $x$ have $\text{nat } [x] \geq a \text{ nat } [x] < b$ by linarith+
  ultimately show $\?thesis$ by force
lemma nat-sum-has-integral-floor:
fixes f :: nat ⇒ 'a :: banach
assumes mn: m < n
shows ((λx. f (nat ⌈x⌉)) has-integral sum f {m..<n}) {real m..real n}
proof –
define D where D = (λi. {real i..real (Suc i)}) ' {m..<n}
have D: D division-of {m..n}
using Union-atLeastAtMost-real-of-nat[OF mn] by (simp add: division-of-def D-def)
have ((λx. f (nat ⌈x⌉)) has-integral (∑X∈D. f (nat ⌈Inf X⌉))) {real m..real n}
proof (rule has-integral-combine-division)
fix X assume X: X ∈ D
have nat ⌈x⌉ = nat ⌈Inf X⌉ if x ∈ X − {Sup X} for x
using that X by (auto simp: D-def nat-eq-iff floor-eq-iff)
hence ((λx. f (nat ⌈x⌉)) has-integral f (nat ⌈Inf X⌉)) X ⟨→
((λx. f (nat ⌈Inf X⌉)) has-integral f (nat ⌈Inf X⌉)) X using X
by (intro has-integral-spike-eq[of {Sup X}]) auto
also from X have ... using has-integral-const-real[of f (nat ⌈Inf X⌉) ⌈Inf X⌉ Sup X]
by (auto simp: D-def)
finally show ((λx. f (nat ⌈x⌉)) has-integral f (nat ⌈Inf X⌉)) X .
qed

lemma nat-sum-has-integral-ceiling:
fixes f :: nat ⇒ 'a :: banach
assumes mn: m < n
shows ((λx. f (nat ⌈x⌉)) has-integral sum f {m..<n}) {real m..real n}
proof –
define D where D = (λi. {real i..real (Suc i)}) ' {m..<n}
have D: D division-of {m..n}
using Union-atLeastAtMost-real-of-nat[OF mn] by (simp add: division-of-def D-def)
have ((λx. f (nat ⌈x⌉)) has-integral (∑X∈D. f (nat ⌈Sup X⌉))) {real m..real n}
proof (rule has-integral-combine-division)
fix X assume X: X ∈ D
have nat ⌈x⌉ = nat ⌈Sup X⌉ if x ∈ X − {Inf X} for x
using that X by (auto simp: D-def nat-eq-iff ceiling-eq-iff)
hence ((λx. f (nat ⌈x⌉)) has-integral f (nat ⌈Sup X⌉)) X ⟨→
((λx. f (nat ⌈Sup X⌉)) has-integral f (nat ⌈Sup X⌉)) X using X
by (intro has-integral-spike-eq[of ⌈Inf X⌉]) auto
also from X have ... using has-integral-const-real[of f (nat ⌈Sup X⌉) ⌈Inf X⌉ Sup X]
by (auto simp: D-def)
finally show \((\lambda x. f (\text{nat } [x])) \text{ has-integral } f (\text{nat } [\text{Sup } X])) X\).
qed

also have \((\sum X \in D. f (\text{nat } [\text{Sup } X])) = (\sum k \in \{m..n\}. f (\text{Suc } k))\)
unfolding D-def by (subst sum.reindex) (auto simp: inj-on-def nat-add-distrib)
also have \(\ldots = (\sum k \in \{m..n\}. f k)\)
by (intro sum.reindex-bij-witness[of - \lambda x. \text{ x - 1 Suc}]) auto
finally show \(?thesis\).
qed

lemma zeta-partial-sum-le:
fixes x :: real and m :: nat
assumes x: \(x > 0\) and m: \(m > 0\)
shows \((\sum _{n=1}^m. \text{real } n \text{ powr } (x - 1)) \leq {\text{real } m \text{ powr } x} / x\)
proof
consider m = 0 | m = 1 | m > 1 by force
thus \(?thesis\)
proof cases
  assume m: \(m > 1\)
hence \(\{1..m\} = \text{insert } 1 \{1..<m\}\) by auto
  also have \((\sum k \in \ldots. \text{real } k \text{ powr } (x - 1)) = 1 + (\sum k \in \{1..<m\}. \text{real } k \text{ powr } (x - 1))\)
  by simp
  also have \((\sum k \in \{1..<m\}. \text{real } k \text{ powr } (x - 1)) \leq \text{real } m \text{ powr } x / x - 1 / x\)
  proof (rule has-integral-le)
    show \((\lambda t. (\text{nat } [t]) \text{ powr } (x - 1)) \text{ has-integral } (\sum n \in \{1..<m\}. n \text{ powr } (x - 1))) \{\text{real } 1..m\}\)
    using m by (intro nat-sum-has-integral-ceiling) auto
  next
  have \((\lambda t. t \text{ powr } (x - 1)) \text{ has-integral } (\text{real } m \text{ powr } x / x - \text{real } 1 \text{ powr } x / x)\)
    \{\text{real } 1..\text{real } m\}\)
  by (intro fundamental-theorem-of-calculus)
  (insert x m, auto simp flip: has-real-derivative-iff-has-vector-derivative
    intro!: derivative-eq-intros)
  thus \((\lambda t. t \text{ powr } (x - 1)) \text{ has-integral } (\text{real } m \text{ powr } x / x - 1 / x)\) \{\text{real } 1..\text{real } m\}\)
  by simp
qed (insert x, auto intro!: powr-mono2')
also have \(1 + (\text{real } m \text{ powr } x / x - 1 / x) \leq \text{real } m \text{ powr } x / x\)
  using x by (simp add: field-simps)
finally show \(?thesis\) by simp
qed (use assms in auto)
qed

lemma zeta-partial-sum-le':
fixes x :: real and m :: nat
assumes x: \(x > 0\) and m: \(m > 0\)
shows \((\sum _{n=1}^m. \text{real } n \text{ powr } (x - 1)) \leq m \text{ powr } x * (1 / x + 1 / m)\)
proof (cases $x > 1$)
case False
with assms have $(\sum n=1..m. \text{real } n \text{ powr } (x - 1)) \leq m \text{ powr } x / x$
  by (intro zeta-partial-sum-le) auto
also have $\ldots \leq m \text{ powr } x * (1 / x + 1 / m)$
  using assms by (simp add: field-simps)
finally show $?thesis$.
next
proof (cases $x$)
  case True
  have $(\sum n\in\{1..m\}. \text{ real } n \text{ powr } (x - 1)) = (\sum n\in insert m \{0..<m\}. \text{ real } n \text{ powr } (x - 1))$
  by (intro sum mono-neutral-left) auto
  also have $\ldots = m \text{ powr } (x - 1) + (\sum n\in\{0..<m\}. \text{ real } n \text{ powr } (x - 1))$ by simp
proof (rule has-integral-le)
  show $(\lambda t. \text{real } \lfloor t \rfloor \text{ powr } (x - 1)) \text{ has-integral } (\text{real } m \text{ powr } (x - 1))$ \{real 0..m\}
    using m by (intro intro nat-sum-has-integral-floor) auto
next
  proof (cases t)
    case T
    with (x > 1) show real (\lfloor t \rfloor) \text{ powr } (x - 1) \leq t \text{ powr } (x - 1)
      by (cases t = 0) (auto intro: powr-mono2)
  qed
  also have $m \text{ powr } (x - 1) + m \text{ powr } x / x = m \text{ powr } x * (1 / x + 1 / m)$
    using m x by (simp add: powr-diff field-simps)
finally show $?thesis$ by simp
qed

lemma natfun-bigo-1E:
  assumes (f :: nat \Rightarrow \cdot) \in O(\lambda x\cdot 1)
  obtains $C$ where $C \geq lb \land n. \text{ norm } (f n) \leq C$
proof
  from assms obtain $C N$ where $\forall n\geq N. \text{ norm } (f n) \leq C$
    by (auto elim!: landau-o.bigE simp: eventually-at-top-linorder)
hence $*: \text{ norm } (f n) \leq \text{Max } (\{C, lb\} \cup (\text{norm } \cdot f \cdot \{..<N\}))$ for n
    by (cases n \geq N) (subst Max-ge_iff; force simp: image_iff)+
moreover have $\text{Max } (\{C, lb\} \cup (\text{norm } \cdot f \cdot \{..<N\})) \geq lb$
    by (intro Max.coboundedI) auto
ultimately have $\text{Max } (\{C, lb\} \cup (\text{norm } \cdot f \cdot \{..<N\})) \geq lb$
    by (intro Max.coboundedI) auto
finally show $?thesis$ using that by blast
qed

lemma natfun-bigo-iff-Bseq: $f \in O(\lambda x\cdot 1) \leftrightarrow Bseq f$
proof
  assume Bseq f.
  then obtain $C$ where $C > 0 \land n. \text{ norm } (f n) \leq C$ by (auto simp: Bseq_def)
then $f \in O(\lambda x\cdot 1)$ by (intro bigO[of - C]) auto

19
next
assume \( f \in O(\lambda^n, 1) \)
from natfun-bigo-1E[of this, where \( lb = 1 \)] obtain \( C \) where \( C \geq 1 \wedge n \). norm \( (f \ n) \leq C \)
by auto
thus \( Bseq \ f \) by (auto simp: Bseq-def intro!: exI[of - \( C \)])
qed

lemma enn-decreasing-sum-le-set-nn-integral:
fixes \( f \) :: real
assumes decreasing: \( \forall x y. \ 0 \leq x \Longrightarrow x \leq y \Longrightarrow f \ y \leq f \ x \)
shows \( (\sum n. f \ (real \ (Suc \ n))) \leq set-nn-integral \ lborel \ \{0..\} \ f \)
proof
have \( (\sum n. f \ (Suc \ n)) = \)
\( \sum n. f ^+ x \{real <..real (Suc \ n)\}. (f \ Suc \ n) \ lborel \)
by (subst nn-integral-cmult-indicator) auto
also have \( \text{nat} \ [x] = Suc \ n \) if \( x \in \{real <..real (Suc \ n)\} \) for \( x \ n \)
using that by (auto simp: nat-eq-iff ceiling-eq-iff)
hence \( (\sum n. \int ^+ x \{real <..real (Suc \ n)\}. (f \ Suc \ n) \ lborel) = \)
\( \sum n. \int ^+ x \{real <..<real (Suc \ n)\}. (f \ real \ \{nat \ [x]\}) \ lborel \)
by (intro suminf-cong nn-integral-cong) (auto simp: indicator-def)
also have \( \ldots = \int ^+ x \{\bigcup i. \{real i..<real (Suc \ i)\}\}. (f \ real \ [x..real]) \ lborel \)
by (auto simp: disjoint-family-on-def)
(auto simp: indicator-def intro!: decreasing)
finally show \( \text{thesis} \).
qed

lemma abs-summable-on-uminus-iff:
\((\lambda x. -f \ x) \ abs-summable-on A \longleftrightarrow f \ abs-summable-on A\)
by (simp add: abs-summable-on-def)

lemma abs-summable-on-cmult-right-iff:
fixes \( f \) :: 'a \Rightarrow 'b :: \{banach, real-normed-field, second-countable-topology\}
assumes \( c \neq 0 \)
shows \( (\lambda x. c * f \ x) \ abs-summable-on A \longleftrightarrow f \ abs-summable-on A \)
by (simp add: abs-summable-on-allow_def assms)

lemma abs-summable-on-cmult-left-iff:
fixes \( f \) :: 'a \Rightarrow 'b :: \{banach, real-normed-field, second-countable-topology\}
assumes \( c \neq 0 \)
shows \( (\lambda x. f \ x * c) \ abs-summable-on A \longleftrightarrow f \ abs-summable-on A \)
by (simp add: abs-summable-on-allow_def assms)

lemma decreasing-sum-le-integral:
fixes \( f \) :: real \Rightarrow real
assumes nonneg: \( \forall x. x \geq 0 \Longrightarrow f \ x \geq 0 \)
assumes decreasing: \( \forall x y. 0 \leq x \Longrightarrow x \leq y \Longrightarrow f \ y \leq f \ x \)
assumes integral: \(f\) has-integral 1 \(\{0\}
\)
shows  summable \((\lambda i. f\ (\text{real}\ (\text{Suc}\ i)))\) and suminf \((\lambda i. f\ (\text{real}\ (\text{Suc}\ i)))\) \(\leq I\)
proof –
have \([\text{simp}]\): \(I \geq 0\)
  by (intro has-integral-nonneg \([\text{OF integral]}\) nonneg) auto

have \((\sum n. \text{ennreal}\ (f\ (\text{Suc}\ n))) = \)
  \((\sum n. \int^+ x \in \{\text{real}\ n<\ldots\text{real}\ (\text{Suc}\ n)\}. \text{ennreal}\ (f\ (\text{Suc}\ n)) \partial\text{borel})\)
  by (subst \(\text{nn-integral-cmult-indicator}\) auto
also have \(\text{nat} [x] = \text{Suc}\ n\ \text{if} \ x \in \{\text{real}\ n<\ldots\text{real}\ (\text{Suc}\ n)\}\ \text{for} \ x\ n\)
  using that by (auto simp: nat-eq-iff ceiling-eq-iff)

hence \((\sum n. \int^+ x \in \{\text{real}\ n<\ldots\text{real}\ (\text{Suc}\ n)\}. \text{ennreal}\ (f\ (\text{Suc}\ n)) \partial\text{borel}) = \)
  \((\sum n. \int^+ x \in \{\text{real}\ n<\ldots\text{real}\ (\text{Suc}\ n)\}. \text{ennreal}\ (f\ (\text{real}\ (\text{nat} [x]))) \partial\text{borel})\)
  by (intro \(\text{suminf-cong nn-integral-cong}\) \(\text{auto simp: indicator-def}\)
also have \(\ldots = (\int^+ x \in (\bigcup i. \{\text{real}\ i<\ldots\text{real}\ (\text{Suc}\ i)\}). \text{ennreal}\ (f\ (\text{nat} [x::real])) \partial\text{borel})\)
  by (subst \(\text{nn-integral-disjoint-family}\)
    (auto simp: disjoint-family-on-def intro\!: measurable-completion)
also have \(\ldots \leq (\int^+ x \in \{0\}. \text{ennreal}\ (f\ x) \partial\text{borel})\)
  by (intro \(\text{nn-integral-mono}\) \(\text{auto simp: indicator-def nonneg intro\!: decreasing}\)
also have \(\ldots = (\int^+ x. \text{ennreal} (\text{indicat-real} \{0\} \ x * f\ x) \partial\text{borel})\)
  by (intro \(\text{nn-integral-cong}\) \(\text{auto simp: indicator-def}\)
also have \(\ldots = \text{ennreal} I\)
using \(\text{nn-integral-has-integral-lebesgue} [\text{OF nonneg integral}]\) by (auto simp: nonneg)

finally have \(*: (\sum n. \text{ennreal} (f\ (\text{Suc}\ n))) \leq \text{ennreal} I .\)
from \(*\) show summable: summable \((\lambda i. f\ (\text{real}\ (\text{Suc}\ i)))\)
  by (intro summable-suminf-not-top) \(\text{auto simp: top-unique intro: nonneg}\)
note \(*\)
also from summable have \((\sum n. \text{ennreal} (f\ (\text{Suc}\ n))) = \text{ennreal} (\sum n. f\ (\text{Suc}\ n))\)
  by (subst \(\text{suminf-ennreal2}\)) \(\text{auto simp: o-def nonneg}\)
finally show \((\sum n. f\ (\text{real}\ (\text{Suc}\ n))) \leq I\) by (subst \(\text{asm}\) ennreal-le-iff) auto
qed

lemma decreasing-sum-le-integral:\
fixes \(f\ : \text{real} \Rightarrow \text{real}\)
assumes \(\forall x. x \geq 0 \Rightarrow f\ x \geq 0\)
assumes \(\forall x y. 0 \leq x \Rightarrow x \leq y \Rightarrow f\ y \leq f\ x\)
assumes \((f\ \text{has-integral} 1) \{0\} .\)
shows  summable \((\lambda i. f\ (\text{real}\ i))\) and suminf \((\lambda i. f\ (\text{real}\ i))\) \(\leq f\ 0 + I\)
proof –
have summable \((\lambda i. f\ (\text{real}\ (\text{Suc}\ i)))\)
  using decreasing-sum-le-integral \([\text{OF assms}]\) \(\text{by simp add: o-def}\)
thus \(*:\) summable \((\lambda i. f\ (\text{real}\ i))\) \(\text{by (subst} \ (\text{asm}) \ \text{summable-Suc-iff})\)
have \((\sum n. f\ (\text{real}\ (\text{Suc}\ n))) \leq I\) by (intro decreasing-sum-le-integral assms)
thus suminf \((\lambda i. f\ (\text{real}\ i))\) \(\leq f\ 0 + I\)
  using \(*\) \(\text{by (subst} \ (\text{asm}) \ \text{suminf-split-head})\) auto
qed

21
lemma of-nat-powr-neq-1-complex [simp]:
assumes \( n > 1 \) \( Re \, s \neq 0 \)
shows of-nat \( n \) powr \( s \) \( \neq (1 :: complex) \)
proof –
  have \( \text{norm} \ (\text{of-nat} \ n \ \text{powr} \ s) = \text{real} \ n \ \text{powr} \ Re \ s \)
    by (simp add: norm-powr-real-powr)
  also have \( \ldots \neq 1 \)
    using assms by (auto simp: powr-def)
finally show ?thesis by auto
qed

lemma fds-logderiv-completely-multiplicative:
fixes \( f :: 'a \cdot \{\text{real-normed-field}\} \) \( \text{fds} \)
assumes \( \text{completely-multiplicative-function} \ (\text{fds-nth} \ f) \) \( \text{fds-nth} \ f \ 1 \neq 0 \)
shows \( \text{fds-deriv} \ f / f = - \text{fds} \ (\lambda n. \ \text{fds-nth} \ f \ n \ * \ \text{mangoldt} \ n) \)
proof –
  have \( \text{fds-deriv} \ f / f = - \text{fds} \ (\lambda n. \ \text{fds-nth} \ f \ n \ * \ \text{mangoldt} \ n) / f \)
    using \( \text{completely-multiplicative-fds-deriv[of fds-nth f]} \)
    assms by simp
  also have \( \ldots = - \text{fds} \ (\lambda n. \ \text{fds-nth} \ f \ n \ * \ \text{mangoldt} \ n) \)
    using assms by (simp add: divide-fds-def fds-right-inverse)
finally show ?thesis .
qed

lemma fds-nth-logderiv-completely-multiplicative:
fixes \( f :: 'a \cdot \{\text{real-normed-field}\} \) \( \text{fds} \)
assumes \( \text{completely-multiplicative-function} \ (\text{fds-nth} \ f) \)
\( \text{fds-nth} \ f \ 1 \neq 0 \)
shows \( \text{fds-nth} \ (\text{fds-deriv} \ f / f) \ ) n = - \text{fds-nth} \ f \ n \ * \ \text{mangoldt} \ n \)
using assms by (subst fds-logderiv-completely-multiplicative) (simp-all add: fds-nth-fds')

lemma eval-fds-logderiv-completely-multiplicative:
fixes \( s :: 'a :: \{\text{dirichlet-series}\} \) and \( l :: 'a \) and \( f :: 'a \) \( \text{fds} \)
defines \( h \equiv \text{fds-deriv} \ f / f \)
assumes \( \text{completely-multiplicative-function} \ (\text{fds-nth} \ f) \) and \( \{\text{simp} \}: \text{fds-nth} \ f \ 1 \neq 0 \)
\( \text{assumes} \ s \cdot 1 > \text{abs-conv-abscissa} \ f \)
shows \( \lambda p. \ \text{of-real} \ (\text{ln} \ (\text{real} \ p)) * (1 / (1 - \text{fds-nth} \ f \ p \ / \ \text{nat-power} \ p \ s - 1)) \)
abs-summable-on \( \{p. \ \text{prime} \ p\} \)
and \( \text{eval-fds} \ h \ ) s = - (\sum p \mid \text{prime} \ p. \ \text{of-real} \ (\text{ln} \ (\text{real} \ p)) * (1 / (1 - \text{fds-nth} \ f \ p \ / \ \text{nat-power} \ p \ s - 1))) \)
(is \( \text{?th1} \))
(is \( \text{?th2} \))
proof –
  let \( ?P = \{p::\text{nat}. \ \text{prime} \ p\} \)
interpret \( f: \text{completely-multiplicative-function} \) \( \text{fds-nth} \ f \) by \( \text{fact} \)
have \( \text{fds-abs-converges} \ h \ ) s
  using \( \text{abs-conv-abscissa-completely-multiplicative-log-deriv[of assms(2)]} \)
  assms by (intro fds-abs-converges) auto
hence \( \lambda n. \ \text{fds-nth} \ h \ n / \ \text{nat-power} \ n \ s \) \( \text{abs-summable-on} \ UNIV \)
  by (auto simp: h-def fds-abs-converges-altdef)

note *
also have \((\lambda n. fds-nth h n / \text{nat-power } n s) \text{ abs-summable-on } \text{UNIV} \iff (\lambda x. -fds-nth f x * \text{mangoldt } x / \text{nat-power } x s) \text{ abs-summable-on } \text{Collect primepow}\)

unfolding \text{h-def using } \text{fds-nth-logderiv-completely-multiplicative[\text{OF assms}(2)]}\n
by (intro abs-summable-on-cong-neutral) (auto simp: \text{fds-nth-fds mangoldt-def})

finally have \(\text{sum1}: (\lambda x. -fds-nth f x * \text{mangoldt } x / \text{nat-power } x s) \text{ abs-summable-on } \text{Collect primepow}\)

by (rule abs-summable-on-subset) auto

also have \(?this \iff (\lambda (p,k). -(\text{fds-nth f } p / \text{nat-power } p s) \sim \text{Suc } k * \text{of-real } (\ln (\text{real } p)))\)

abs-summable-on \((?P \times \text{UNIV})\)

unfolding case-prod-unfold

by (intro abs-summable-on-cong, subst mangoldt-primepow)

(auto simp: f.mutil f.power nat-power-mult-distrib nat-power-power-left power-divide
  dest: prime-gt-1-nat)

finally have \(\text{sum2}: \ldots \).

have \(\text{sum4}: \text{summable } (\lambda n. (\text{norm } (\text{fds-nth f } p / \text{nat-power } p s)) \sim \text{Suc } n) \text{ if } p:\)

prime \(p\) for \(p\)

proof -

  have \(\text{summable } (\lambda n. [\ln (\text{real } p)] * (\text{norm } (\text{fds-nth f } p / \text{nat-power } p s)) \sim \text{Suc } n)\)

  using \(p\) \text{abs-summable-on-Sigma-project2[\text{OF assms2}, of } p\) unfolding \text{abs-summable-on-nat-iff'}

  by (simp add: norm-power norm-mult norm-divide mult-ac del: power-Suc)

  thus \(?thesis\) by (rule summable-mult-D) (insert p, auto dest: prime-gt-1-nat)

qed

have \(\text{sums}: (\lambda n. (\text{fds-nth f } p / \text{nat-power } p s) \sim \text{Suc } n) \text{ sums}\)

\((-1 / (1 - \text{fds-nth f } p / \text{nat-power } p s) - 1) \text{ if } p: \text{prime } p\text{ for } p:: \text{nat}\)

proof -

  from \(\text{sum4}[\text{OF } p]\) have \(\text{norm } (\text{fds-nth f } p / \text{nat-power } p s) < 1\)

  unfolding \text{summable-Suc-iff} by (simp add: summable-geometric-iff)

  from \text{geometric-sums[OF this]} show \(?thesis\) by (subst \text{sums-Suc-iff}) auto

qed

have \(\text{eq}: (\sum a.k. - ((\text{fds-nth f } p / \text{nat-power } p s) \sim \text{Suc } k * \text{of-real } (\ln (\text{real } p))))\)

\(=\)

\((-\text{of-real } (\ln (\text{real } p)) * (1 / (1 - \text{fds-nth f } p / \text{nat-power } p s) - 1))\)

if \(p: \text{prime } p\text{ for } p\)

proof -

  have \((\sum a.k. - ((\text{fds-nth f } p / \text{nat-power } p s) \sim \text{Suc } k * \text{of-real } (\ln (\text{real } p)))) =\)

  \((\sum a.k. (\text{fds-nth f } p / \text{nat-power } p s) \sim \text{Suc } k) * \text{of-real } (\ln (\text{real } p)))\)

  using \(\text{sum4}[\text{of } p]\)

  by (subst \text{infsums-cmult-left [symmetric]})

  (auto simp: abs-summable-on-nat-iff' norm-power simp del: power-Suc)

  also have \((\sum a.k. (\text{fds-nth f } p / \text{nat-power } p s) \sim \text{Suc } k) =\)

23
\begin{align*}
& \left(1 / \left(1 - \text{fds-nth } f \ p / \text{nat-power } p \ s\right) - 1\right) \text{ using sum4}[OF p] \\
& \text{by (subst infsetsum-nat')}
\end{align*}

\text{auto simp: sums-iff abs-summable-on-nat-iff' norm-power simp del: power-Suc)

finally show ?thesis by (simp add: mult-ac)
qed

have \begin{align*}
& \text{sum3: } (\lambda x. \sum_{a} y. - ((\text{fds-nth } f \ x / \text{nat-power } x \ s) \sim \text{Suc } y \ast \text{of-real } (\ln \text{real } x))) \\
& \text{abs-summable-on } \{ \text{p. prime } p\}
\end{align*}

\text{using sum2 by (rule abs-summable-on-Sigma-project1') auto}
also have ?this \iff (\lambda p. - (\text{of-real } (\ln \text{real } p) \ast \\
\left(1 / \left(1 - \text{fds-nth } f \ p / \text{nat-power } p \ s\right) - 1\right))) \text{abs-summable-on } \{ \text{p. prime } p\}

\text{by (intro abs-summable-on-cong eq) auto}

finally show ?th1.

have \begin{align*}
& \text{eval-fds } h \ s = (\sum_{a} n. \text{fds-nth } h \ n / \text{nat-power } n \ s) \\
& \text{using * unfolding eval-fds-def by (subst infsetsum-nat') auto}
\end{align*}
also have \begin{align*}
& \text{... } = (\sum_{a} n \in \{ \text{n. primepow } n\}. -\text{fds-nth } f \ n \ast \text{mangoldt } n / \text{nat-power } n \ s)
\end{align*}

\text{unfolding h-def using fds-nth-logderiv-completely-multiplicative}[OF assms(2)]
\text{by (intro infsetsum-cong-neutral) (auto simp: fds-nth-fds mangoldt-def)}
also have \begin{align*}
& \text{... } = (\sum_{a}(p,k) \in (?P \times \text{UNIV}). -\text{fds-nth } f \ (p \sim \text{Suc } k) \ast \text{mangoldt } (p \sim \text{Suc } k) / \\
& \text{nat-power } (p \sim \text{Suc } k) \ s)
\end{align*}

\text{using bij-betw-primepows unfolding case-prod-unfold}
\text{by (intro infsetsum-reindex-bij-betw [symmetric])}
also have \begin{align*}
& \text{... } = (\sum_{a}(p,k) \in (?P \times \text{UNIV}). -((\text{fds-nth } f \ p / \text{nat-power } p \ s) \sim \text{Suc } k) \ast \text{of-real } (\ln \text{real } p))
\end{align*}
\text{by (intro infsetsum-cong)}
\text{(auto simp: f.mult f.power mangoldt-def aprime divisor-prime-power ln-realpow prime-qt-0-nat)

\text{nat-power-power-left divide-simps simp del: power-Suc)

also have \begin{align*}
& \text{... } = (\sum_{a} p \mid \text{prime } p. \sum_{a} k. \\
& - ((\text{fds-nth } f \ p / \text{nat-power } p \ s) \sim \text{Suc } k) \ast \text{of-real } (\ln \text{real } p))
\end{align*}
\text{using sum2 by (subst infsetsum-Times) (auto simp: case-prod-unfold)}
also have \begin{align*}
& \text{... } = (\sum_{a} p \mid \text{prime } p. - (\text{of-real } (\ln \text{real } p)) \ast \\
& \left(1 / \left(1 - \text{fds-nth } f \ p / \text{nat-power } p \ s\right) - 1\right))
\end{align*}
\text{using eq by (intro infsetsum-cong) auto}
finally show ?th2 by (subst (asm) infsetsum-uminus)
qed

\text{lemma eval-fds-logderiv-zeta:}
\text{assumes Re } s > 1
\text{shows } (\lambda p. \text{of-real } (\ln \text{real } p) / (p \text{ powr } s - 1))
\text{abs-summable-on } \{ \text{p. prime } p\} \{\text{is ?th1}\)
\[ \text{deriv } \zeta(s) / \zeta(s) = -\left(\sum_{p \mid 	ext{prime } p} \text{of-real } (\ln(\text{real } p)) / (p^\text{powr } s - 1)\right) \] (is ?th2)

**proof** –

**have** *: completely-multiplicative-function (fds-nth fds-zeta :: - \Rightarrow complex)

**by** standard auto

**note** abscissa = le-less-trans[OF abs-conv-abscissa-completely-multiplicative-log-deriv[OF *]]

**have** \((\lambda p. \text{ln } (\text{real } p) \ast (1 / (1 - \text{fds-nth } \text{fds-zeta } p / p^\text{powr } s - 1)))\) abs-summable-on \(\{p. \text{prime } p\}\)

**using** eval fds-logderiv-completely-multiplicative[OF * , of s] assms by auto

**also have** ?this \(\longleftrightarrow (\lambda p. \text{ln } (\text{real } p) / (p^\text{powr } s - 1))\) abs-summable-on \(\{p. \text{prime } p\}\)

**by** (intro abs-summable-on-cong) (auto simp: fds-nth-zeta divide-simps dest: prime-gt-1-nat)

finally **show** ?th1.

**qed**

**lemma** sums-logderiv-zeta:

**assumes** Re \(s > 1\)

**shows** \((\lambda p. \text{if prime } p \text{ then of-real } (\ln(\text{real } p)) / (\text{of-nat } p \text{ powr } s - 1) \text{ else } 0)\) sums

\[-(\text{deriv } \zeta(s) / \zeta(s)) \] (is ?f sums -)

**proof** –

**note** * = eval fds-logderiv-zeta[OF assms]

**from** sums-infsetsum-nat[OF *(1)] \(\text{and } *(2)\) **show** ?thesis by simp

**qed**

**lemma** range-add-nat: range \((\lambda n. n + c) = \{(c::nat)\}..\)

**using** Nat.le-imp-diff-is-add by auto

**lemma** abs-summable-hurwitz-zeta:
assumes \( Re \ s > 1 \ a + \text{real} \ b > 0 \)
shows \( (\lambda n. 1 / (\text{of-nat} \ n + a) \ \text{powr} \ s) \ \text{abs-summable-on} \ \{b..\} \)
proof
  from \text{assms} \ have \text{summable} \ (\lambda n. \text{cmod} (1 / (\text{of-nat} \ (n + b) + a) \ \text{powr} \ s))
  using \text{summable-hurwitz-zeta-real}[(\text{of} \ Re \ s \ a + b)]
  by (auto simp: \text{norm-divide} \ \text{powr-minus} \ \text{field-simps} \ \text{norm-powr-real-powr})
hence \( (\lambda n. 1 / (\text{of-nat} \ (n + b) + a) \ \text{powr} \ s) \ \text{abs-summable-on} \ \text{UNIV} \)
  by (auto simp: \text{abs-summable-on-nat-iff} \text{add-ac})
also have \( \lambda n. a + b \)
  by (rule \text{abs-summable-on-reindex-iff}) \text{auto}
also have range \( (\lambda n. n + b) \) \( \{b..\} \) by (rule \text{range-add-nat})
finally show \( \lambda n. n + b \)
qed

lemma hurwitz-zeta-nat-conv-infsetsum:
assumes \( a > 0 \) and \( Re \ s > 1 \)
shows \( \text{hurwitz-zeta} \ (\text{real} \ a) \ s = (\sum a n \cdot \text{of-nat} \ (n + a) \ \text{powr} \ s) \)
\( \text{hurwitz-zeta} \ (\text{real} \ a) \ s = (\sum a n \in \{a..\} \cdot \text{of-nat} \ n \ \text{powr} \ s) \)
proof
  have \( \text{hurwitz-zeta} \ (\text{real} \ a) \ s = (\sum a n \cdot \text{of-nat} \ (n + a) \ \text{powr} \ s) \)
  using \text{assms} by (subst \text{hurwitz-zeta-conv-saminf}) \text{auto}
  also have \( \ldots = (\sum a n \cdot \text{of-nat} \ (n + a) \ \text{powr} \ s) \)
  using \text{assms} by (intro \text{infsetsum-nat'} [\text{symmetric}]) (auto simp: \text{powr-minus} \ \text{field-simps})
finally show \( \text{hurwitz-zeta} \ (\text{real} \ a) \ s = (\sum a n \cdot \text{of-nat} \ (n + a) \ \text{powr} \ s) \)
  by (rule \text{infsetsum-reindex} [\text{symmetric}] \text{auto})
  also have \( \ldots = (\sum a n \in \text{range} \ \lambda n. n + a) \cdot \text{of-nat} \ n \ \text{powr} \ s) \)
  by (rule \text{range-add-nat})
finally show \( \text{hurwitz-zeta} \ (\text{real} \ a) \ s = (\sum a n \in \{a..\} \cdot \text{of-nat} \ n \ \text{powr} \ s) \)
qed

lemma pre-zeta-bound:
assumes \( 0 < \text{Re} \ s \) and \( a: a > 0 \)
shows \( \text{norm} \ (\text{pre-zeta} \ a \ s) \leq (1 + \text{norm} \ s / \text{Re} \ s) / 2 \ a \ \text{powr} \ \text{Re} \ s \)
proof
  let \( \lambda f = \lambda x. -(s \ast (x + a) \ \text{powr} (-1 \ast s)) \)
  let \( \lambda g = \lambda x. \text{norm} \ s \ast (x + a) \ \text{powr} (-1 - \text{Re} \ s) \)
  let \( \lambda g = \lambda x. -(\text{norm} \ s / \text{Re} \ s \ast (x + a) \ \text{powr} (-\text{Re} \ s)) \)

define \( R \) where \( R = \text{EM-remainder} \ 1 \ ?f \ 0 \)

have \( \text{simp:} -\text{Re} \ s - 1 = -1 - \text{Re} \ s \) \text{by (simp add: algebra-simps)}

have \( |\text{frac} \ x - 1 / 2| \leq 1 / 2 \) \text{for} \( x::\text{real} \) \text{unfolding frac-def}
  by linarith

hence \( |\text{pbernopoly} \ (\text{Suc} \ 0) \ x| \leq 1 / 2 \) \text{for} \( x \)
  by (simp add: pbernopoly-def bernpoly-def)

moreover have \( (\lambda b. \text{cmod} \ s \ast (b + a) \ \text{powr} -\text{Re} \ s / \text{Re} \ s) \ |\leq 0 \) \text{at-top}
  using \( \text{Re} \ s > 0 \) \( a > 0 \) \text{by real-asympt}

ultimately have \( *: \forall x. \ x \geq \text{real} \ 0 \ |\leq \text{norm} \ (\text{EM-remainder} \ 1 \ ?f \ (\text{int} \ x)) \leq \)
\[(1 / 2) / \text{fact} \ 1 \ast (-g \text{ (real } x))\]
using \(\langle a > 0, \ Re s > 0, \rangle\)
by (intro norm-EM-remainder-le-strong-nat [where \(g' = \ ?g' \text{ and } Y = \{\}]\)
(auto intro!: continuous-intros derivative-eq-intros
simp: field-simps norm-mult norm-powr-real-powr add-eq-0-iff)
have \(R: \norm R \leq \norm s / (2 \ast Re s) \ast a \powr Re s\)
unfolding R-def using spec[OF \(\ast, \ of \ 0\)] by simp
from assms have \(\text{pre-zeta } a \ s = a \powr -s / 2 + R\)
by (simp add: pre-zeta-def pre-zeta-aux-def R-def)
also have \(\norm \ldots \leq a \powr -Re s / 2 + \norm s / (2 \ast Re s) \ast a \powr -Re s\)
using \(a\)
by (intro order.trans[OF norm-triangle-ineq add-mono] (auto simp: norm-powr-real-powr)
also have \(\ldots = (1 + \norm s / Re s) / 2 \ast a \powr -Re s\)
by (simp add: field-simps)
finally show \(?thesis\).
qed

lemma \(\text{pre-zeta-bound'}:\)
assumes \(0 < Re s \text{ and } a: a > 0\)
shows \(\norm (\text{pre-zeta } a \ s) \leq \norm s / (Re s \ast a \powr Re s)\)
proof –
from assms have \(\norm (\text{pre-zeta } a \ s) \leq (1 + \norm s / Re s) / 2 \ast a \powr -Re s\)
by (intro pre-zeta-bound) auto
also have \(\ldots = (Re s + \norm s) / 2 / (Re s \ast a \powr Re s)\)
using assms by (auto simp: field-simps power-minus)
also have \(Re s + \norm s \leq \norm s + \norm s \text{ by (intro add-right-mono complex-Re-le-cmod)}\)
also have \((\norm s + \norm s) / 2 = \norm s\) by simp
finally show \(\norm (\text{pre-zeta } a \ s) \leq \norm s / (Re s \ast a \powr Re s)\)
using assms by (simp add: divide-right-mono)
qed

lemma \(\text{deriv-zeta-eq}:\)
assumes \(s: s \neq 1\)
shows \(\text{deriv } zeta s = \text{deriv } (\text{pre-zeta } 1 \ s - 1 / (s - 1)^2)\)
proof –
from s have ev: eventually \((\lambda z. \ z \neq 1) \text{ (nhds } s)\) by (intro t1-space-nhds)
have [derivative-intros]: \(\text{pre-zeta } 1 \ has-field-derivative \ \text{deriv } (\text{pre-zeta } 1 \ s) \ (at \ s)\)
by (intro holomorphic-derivI[of - UNIV] holomorphic-intros) auto
have \((\lambda s. \ \text{pre-zeta } 1 \ s + 1 / (s - 1)) \ has-field-derivative \ \text{deriv } (\text{pre-zeta } 1 \ s - 1 / (s - 1)^2)) \ (at \ s)\)
using s by (auto intro!: derivative-eq-intros simp: power2-eq-square)
also have \(?this \iff (zeta \ has-field-derivative \ \text{deriv } (\text{pre-zeta } 1 \ s - 1 / (s - 1)^2)) \ (at \ s)\)
by (intro has-field-derivative-cong-ev eventually-mono[OF ev])
(auto simp: zeta-def hurwitz-zeta-def)
finally show \(?\text{thesis}\) by (rule DERIV-imp-deriv)

qed

lemma zeta-remove-zero:
  assumes \(\text{Re } s \geq 1\)
  shows \((s - 1) \ast \text{pre-zeta } 1 \ast s + 1 \neq 0\)
proof (cases \(s = 1\))
  case False
  hence \((s - 1) \ast \text{pre-zeta } 1 \ast s + 1 = (s - 1) \ast \text{zeta } s\)
    by (simp add: zeta-def hurwitz-zeta-def divide-simps)
also from False assms have \(\ldots \neq 0\) using zeta-Re-ge-1-nonzero[of \(s\)] by auto
finally show \(?\text{thesis}\).
qed auto

lemma eval-fds-deriv-zeta:
  assumes \(\text{Re } s > 1\)
  shows \(\text{eval-fds (fds-deriv fds-zeta) } s = \text{deriv zeta } s\)
proof
  have \(\text{ev: eventually } (\lambda z. \text{Re z } > 1) (\text{nhds } s)\)
    using assms by (intro eventually-nhds-in-open open-halfspace-Re-gt auto)
from assms have \(\text{eval-fds (fds-deriv fds-zeta) } s = \text{deriv (eval-fds fds-zeta) } s\)
  by (subst eval-fds-deriv) auto
also have \(\ldots = \text{deriv zeta } s\)
  by (intro deriv-cong-ev eventually-mono[OF \(\text{ev}\)]) (auto simp: eval-fds-zeta)
finally show \(?\text{thesis}\).
qed

lemma le-nat-iff': \(x \leq \text{nat } y \leftrightarrow x = 0 \land y \leq 0 \lor \text{int } x \leq y\)
  by auto

lemma sum-upto-plus1:
  assumes \(x \geq 0\)
  shows \(\text{sum-upto } f (x + 1) = \text{sum-upto } f x + f (\text{Suc (nat } [x]))\)
proof
  have \(\text{sum-upto } f (x + 1) = \text{sum } f \{0<..\text{Suc (nat } [x])\}\)
    using assms by (simp add: sum-upto-altdef nat-add-distrib)
also have \(\{0<..\text{Suc (nat } [x])\} = \text{insert (Suc (nat } [x]) \{0<..\text{nat } [x]\}\)
    by auto
also have \(\text{sum } f \ldots = \text{sum-upto } f x + f (\text{Suc (nat } [x]))\)
    by (subst sum.insert) (auto simp: sum-upto-altdef add-AC)
finally show \(?\text{thesis}\).
qed

lemma sum-upto-minus1:
  assumes \(x \geq 1\)
  shows \(\text{sum-upto } f (x - 1) = (\text{sum-upto } f x - f (\text{nat } [x])) :: \text{a :: ab-group-add}\)
  using sum-upto-plus1[of \(x - 1\)] assms by (simp add: algebra-simps nat-diff-distrib)

lemma integral-smallo:
fixes $f$, $g$, $g'$ :: real $\rightarrow$ real
assumes $f \in o(g')$ and filterlim $g$ at-top at-top
assumes $\forall x. \ a \leq x \implies a' \leq x \implies f$ integrable-on $\{a..x\}$
assumes deriv: $\forall x. \ x \geq a \implies (g \ has-field-derivative \ g' \ x) \ (at \ x)$
assumes cont: continuous-on $\{a..\} \ g'$
assumes nonneg: $\forall x. \ x \geq a \implies g' \ x \geq 0$
shows $(\lambda x. \ integral \{a..x\} \ f) \in o(g)$

proof (rule landau-o.smallI)
fix $c :: real$ assume $c \in c > 0$
note [continuous-intros] = continuous-on-subset[OF cont]
define $c' :: real$ where $c' = c / 2$
from $c$ have $c' : c' \in c > 0$ by (simp add: $c'$-def)
from landau-o.smallI[OF assms] this
obtain $b :: real$ where $b : \forall x. \ x \in b \implies norm \ (f \ x) \leq c' \ * \ norm \ (g' \ x)$
unfolding eventually-at-top-linorder by blast
define $b' :: real$ where $b' = \max a \ b$
define $D :: real$ where $D = norm \ (integral \{a..b'\} \ f)$

have filterlim $(\lambda x. \ c' \ * \ g \ x)$ at-top at-top
using $c'$ by (intro filterlim-tendsto-pos-mult-at-top)[OF tendsto-const] assms
hence eventually $(\lambda x. \ c' \ * \ g \ x \geq D - c' \ * \ g \ b')$ at-top
by (auto simp; filterlim-at-top)
thus eventually $(\lambda x. \ norm \ (integral \{a..x\} \ f) \leq c' \ * \ norm \ (g \ x))$ at-top
using eventually-ge-at-top[of $b'$]

proof eventually-elim
  case (elim $x$)
  have $b'$: $a \leq b' \ b \leq b'$ by (auto simp; $b'$-def)
  from $elim \ b'$ have integrable: $(\lambda x. \ |g' \ x|)$ integrable-on $\{b'..x\}$
  by (intro integrable-continuous-real continuous-intros auto)
  have integral $\{a..x\} \ f = integral \{a..b'\} \ f + integral \{b'..x\} \ f$
  using $elim \ b'$ by (intro Henstock-Kurzweil-Integration.integral-combine [symmetric]
assms) auto
  also have norm $\ldots \leq D + norm \ (integral \{b'..x\} \ f)$
  unfolding $D$-def by (rule norm-triangle-ineq)
  also have norm $(integral \{b'..x\} \ f) \leq integral \{b'..x\} \ (\lambda x. \ c' \ * \ norm \ (g' \ x))$
  using $b' \ elim$ assms $c'$ integrable by (intro integral-norm-bound-integral $b'$
assms) auto
  also have $\ldots = c' \ * \ integral \{b'..x\} \ (\lambda x. \ |g' \ x|)$ by simp
  also have integral $\{b'..x\} \ (\lambda x. \ |g' \ x|) = integral \{b'..x\} \ g'$
  using assms $b'$ by (intro integral-cong) auto
  also have $(g' \ has-integral \ (g \ x - g \ b')) \{b'..x\}$ using $b' \ elim$
  by (intro fundamental-theorem-of-calculus)
  (auto simp flip: has-real-derivative-iff-has-vector-derivative
  intro!: has-field-derivative-at-within[OF deriv])
  hence integral $\{b'..x\} \ g' = g \ x - g \ b'$$
  by (simp add: has-integral-iff)
  also have $D + c' \ * (g \ x - g \ b') \leq c' \ * g \ x$
  using $elim$ by (simp add: field-simps $c'$-def)
  also have $\ldots \leq c' \ * norm \ (g \ x)$
lemma integral-bigo:

fixes \( f, g \) :: \( \text{real} \Rightarrow \text{real} \)

assumes \( f \in O(g) \) and \( \text{filterlim} \, g \, \text{at-top} \, \text{at-top} \)

assumes \( \forall a'. \, a \leq a' \Rightarrow a' \leq x \Rightarrow f \, \text{integrable-on} \, \{a'..x\} \)

assumes \( \text{deriv} : \forall x. \, x \geq a \Rightarrow (g \, \text{has-field-derivative} \, g' \, x) \) (at \( x \) within \( \{a..\} \))

assumes \( \text{cont} : \text{continuous-on} \, \{a..\} \, g' \)

assumes \( \text{nonneg} : \forall x. \, x \geq a \Rightarrow g' \, x \geq 0 \)

shows \( (\lambda x. \, \text{integral} \, \{a..x\} \, f) \in O(g) \)

proof

note \([\text{continuous-intros} = \text{continuous-on-subset}[\text{OF cont}]\]

from landau-o.bigE[\text{OF assms(1)}]

obtain \( c, b \) where \( c > 0 \) and \( b : \forall x. \, x \geq b \Rightarrow \text{norm} \, f \, x \leq c \times \text{norm} \, g' \, x \)

unfolding \( \text{eventually-at-top-linorder} \) by \text{metis}

define \( c' \) where \( c' = c / 2 \)

define \( b' \) where \( b' = \max a \, b \)

define \( D \) where \( D = \text{norm} \, (\text{integral} \, \{a..b'\} \, f) \)

have \( \text{filterlim} \, (\lambda x. \, c \times g \, x) \, \text{at-top} \, \text{at-top} \)

using \( c \) by \([\text{intro filterlim-tendsto-pos-mult-at-top}[\text{OF tendsto-const}] \, \text{assms}]\)

hence \( \forall x. \, c \times g \, x \geq D - c \times g \, b' \) \, \text{at-top}

by \( (\text{auto simp: filterlim-at-top}) \)

hence \( \forall x. \, \text{norm} \, (\text{integral} \, \{a..x\} \, f) \leq 2 \times c \times \text{norm} \, (g' \, x) \) \, \text{at-top}

using \( \text{eventually-ge-at-top}[\text{of b']\)}

proof \( \text{eventually-elim} \)

\( \text{case (elim x)} \)

have \( b' : a \leq b' \, b \leq b' \) by \( (\text{auto simp: b'-def}) \)

from \( \text{elim b'} \) have \( \text{integrable}: \forall x. \, \text{|g' \, x|} \, \text{integrable-on} \, \{b'..x\} \)

by \( (\text{intro integrable-continuous-real continuous-intros}) \)

have \( \text{integral} \, \{a..x\} \, f = \text{integral} \, \{a..b'\} \, f + \text{integral} \, \{b'..x\} \, f \)

using \( \text{elim b'} \) by \( (\text{intro Henstock-Kurzweil-Integration.integral-combine [symmetric]} \, \text{assms}) \)

auto

also have \( \text{norm} \ldots \leq D + \text{norm} \, (\text{integral} \, \{b'..x\} \, f) \)

unfolding \( \text{D-def by (rule norm-triangle-ineq)} \)

also have \( \text{norm} \, (\text{integral} \, \{b'..x\} \, f) \leq \text{integral} \, \{b'..x\} \, (\lambda x. \, c \times \text{norm} \, (g' \, x)) \)

using \( b' \, \text{elim assms c integrable by (intro integral-norm-bound-integral b assms) auto} \)

also have \( \ldots = c \times \text{integral} \, \{b'..x\} \, (\lambda x. \, |g' \, x|) \) by \text{simp}

also have \( \text{integral} \, \{b'..x\} \, (\lambda x. \, |g' \, x|) = \text{integral} \, \{b'..x\} \, g' \)

using \( \text{assms b'} \) by \( (\text{intro integral-cong}) \) auto

also have \( (g' \, \text{has-integral} \, (g \, x - g \, b')) \, \{b'..x\} \) \, using \( b' \, \text{elim} \)

by \( (\text{intro fundamental-theorem-of-calculus}) \)

(auto simp flip; has-real-derivative-iff-has-vector-derivative intro!: DERIV-subset[\text{OF deriv}])

qed
hence \( \text{integral \{b',x\}} \) \( g' = g \ x - g \ b' \)

by \((\text{simp add: has-integral-iff})\)

also have \( D + c * (g \ x - g \ b') \leq 2 * c * g \ x \)

using \( \text{elim by (simp add: field-simps c'-def)} \)

also have \( \ldots \leq 2 * c * \text{norm} (g \ x) \)

using \( c \) by \((\text{intro mult-left-mono})\) \(\text{auto}\)

finally show \( ?\text{case by simp}\)

qed

thus \( ?\text{thesis by (rule bigoI)}\)

qed

lemma primepows-le-subset:

assumes \( x: x > 0 \) and \( l: l > 0 \)

shows \( \{(p, i). \text{prime p} \land l \leq i \land \text{real} (p ^ i) \leq x\} \subseteq \{..\text{nat \{root l x\}}\} \times \{..\text{nat \{log 2 x\}}\} \)

proof \( \text{safe}\)

fix \( p i :: \text{nat assume pi: prime p} \ i \geq l \ \text{real} (p ^ i) \leq x \)

have \( \text{real p ^ l} \leq \text{real p ^ i} \) using \( \text{pi x l} \)

by \((\text{intro power-increasing})\) \((\text{auto dest: prime-gt-0-nat})\)

also have \( \ldots \leq x \) using \( \text{pi} \)

by \(\text{simp}\)

finally have \( \text{root l} (\text{real p ^ l}) \leq \text{root l x} \)

using \( \text{x pi l} \) by \((\text{subst real-root-le-iff})\) \(\text{auto}\)

also have \( \text{root l} (\text{real p ^ l}) = \text{real p} \)

using \( \text{pi l} \) by \((\text{subst real-root-pos2})\) \(\text{auto}\)

finally show \( p \leq \text{nat \{root l x\}} \) using \( \text{pi l x} \)

by \((\text{simp add: le-nat-iff le-floor-iff le-log-iff powr-realpow})\)

qed

lemma mangoldt-non-primepow:

\( \neg \text{primepow n} \Rightarrow \text{mangoldt n} = 0 \)

by \((\text{auto simp: mangoldt-def})\)

lemma ln-minus-ln-floor-bigo:

\( (\lambda x. \ln x - \ln (\text{real} \{\text{nat} \{x\}\})) \in O(\lambda-. 1) \)

proof \((\text{intro le-imp-bigo-real[of 1 eventually-mono[of eventually-ge-at-top[of 1]]})\)

fix \( x :: \text{real assume x: x \geq 1} \)

from \( x \) have \( x - \text{real} (\text{nat} \{x\}) \leq 1 \) by \(\text{linarith}\)

from \( x \) have \( \ln x - \ln (\text{real} (\text{nat} \{x\})) \leq (x - \text{real} (\text{nat} \{x\})) / \text{real} (\text{nat} \{x\}) \)

by \((\text{intro ln-diff-le})\) \(\text{auto}\)

also have \( \ldots \leq 1 / 1 \) using \( x * \) by \((\text{intro frac-le})\) \(\text{auto}\)

finally show \( \ln x - \ln (\text{real} (\text{nat} \{x\})) \leq 1 * 1 \) by \(\text{simp}\)

qed \(\text{auto}\)

lemma cos-geD:

assumes \( \text{cos} x \geq \text{cos} a 0 \leq a \ leq \pi - \pi \leq x \leq \pi \)
shows \( x \in \{-a..a\} \)

proof (cases \( x \geq 0 \))
  case True
  with assms show \(?thesis\)
    by (subst (asm) cos-mono-le-eq) auto
next
case False
  with assms show \(?thesis using cos-mono-le-eq[\of a \ -x]\)
    by auto
qed

lemma path-image-part-circlepath-same-Re:
  assumes \( 0 \leq b \ b \leq \pi a = -b \ r \geq 0 \)
  shows \( \text{path-image} (\text{part-circlepath } c \ r \ a \ b) = \text{sphere } c \ r \cap \{ s. \ \Re s \geq \Re c + r \ast \cos a \} \)
proof safe
  fix \( z \)
  assume \( z \in \text{path-image} (\text{part-circlepath } c \ r \ a \ b) \)
  with assms obtain \( t \) where \( t: t \in \{a..b\} \)
    \( \text{z = } c + \text{of-real } r \ast \text{cis } t \)
    by (auto simp: path-image-part-circlepath exp-eq-polar) 
  from \( t \) and assms show \( z \in \text{sphere } c \ r \)
    by (auto simp: dist-norm norm-mult)
  from \( t \) and assms show \( \Re z \geq \Re c + r \ast \cos a \)
    using cos-monotone-0-pi-le \[ of \ t \ b \]
    cos-monotone-minus-pi-0' \[ of a \ t \]
    by (cases \( t \geq 0 \)) (auto intro: mult-left-mono)
next
  fix \( z \)
  assume \( z: z \in \text{sphere } c \ r \ \Re z \geq \Re c + r \ast \cos a \)
  show \( z \in \text{path-image} (\text{part-circlepath } c \ r \ a \ b) \)
  proof (cases \( r = 0 \))
    case False
    with assms have \( r: r > 0 \) by simp
    with \( z \) have \( z\text{-eq: } z = c + r \ast \text{cis } (\text{Arg} (z - c)) \)
      using Arg-eq \[ of \ z - c \]
      by (auto simp: dist-norm exp-eq-polar norm-minus-commute)
    moreover from \( z(2) \) \( r \) assms have \( \cos b \leq \cos (\text{Arg} (z - c)) \)
      by (subst (asm) \( z\text{-eq} \)) auto
    with assms have \( \text{Arg} (z - c) \in \{-b..b\} \)
      using Arg-le-pi\[ of \ z - c \]
      mpi-less-Arg\[ of \ z - c \]
      by (intro cos-geD) auto
    ultimately show \( z \in \text{path-image} (\text{part-circlepath } c \ r \ a \ b) \)
      using assms by (subst path-image-part-circlepath) (auto simp: exp-eq-polar)
  qed (insert assms \( z \), auto simp: path-image-part-circlepath)
  qed

lemma part-circlepath-rotate-left:
  part-circlepath \( c \ r \ (x + a) \ (x + b) = (\lambda z. \ c + \text{cis } x \ast (z - c)) \circ \text{part-circlepath} \ c \ r \ a \ b \)
  by (simp add: part-circlepath-def exp-eq-polar fun-eq-iff 
    linepath-translate-left linepath-translate-right cis-mult add-ac)

lemma part-circlepath-rotate-right:
part-circlepath \( c \) \( (a + x) \) \( (b + x) \) = \((\lambda z. \, c + cîs x * (z - c)) \circ \) part-circlepath
\( c \) \( r a b \)
by (simp add: part-circlepath-def exp-eq-polar fun-eq-iff
linepath-translate-left linepath-translate-right cis-mult add-ac)

**lemma** path-image-semicircle-Re-ge:
assumes \( r \geq 0 \)
shows \( \text{path-image} \ (\text{part-circlepath} \ c \ r \ (-\pi/2) \ (\pi/2)) = \sphere c r \cap \{s. \, \Re s \geq \Re c\} \)
by (subst path-image-part-circlepath-same-Re) (simp-all add: assms)

**lemma** sphere-rotate: \((\lambda z. \, c + cîs x * (z - c)) \)' \( \sphere c r \)
proof safe
fix \( z \) assume \( z \in \sphere c r \)
hence \( z = c + cîs x * (c + cîs (-x) * (z - c) - c) \)
c + cîs (-x) * (z - c) \in \sphere c r
by (auto simp: dist-norm norm-mult norm-minus-commute
cîs-conv-exp exp-minus field-simps norm-divide)
with \( z \) show \( z \in (\lambda z. \, c + cîs x * (z - c)) \)' \( \sphere c r \)
by blast
qed (auto simp: dist-norm norm-minus-commute norm-mult)

**lemma** path-image-semicircle-Re-le:
assumes \( r \geq 0 \)
shows \( \text{path-image} \ (\text{part-circlepath} \ c \ r \ (\pi/2) \ (3/2*\pi)) = \sphere c r \cap \{s. \, \Re s \leq \Re c\} \)
proof
let \( ?f = (\lambda z. \, c + cîs \pi * (z - c)) \)
have \(*: \text{part-circlepath} \ c \ r \ (\pi/2) \ (3/2*\pi) = \text{part-circlepath} \ c \ r \ (\pi + (-\pi/2)) \ (\pi + \pi/2) \)
by simp
have \( \text{path-image} \ (\text{part-circlepath} \ c \ r \ (\pi/2) \ (3/2*\pi)) = ?f \) \( \) \( ?f \) \( \sphere c r \cap \{s. \, \Re c \leq \Re s\} \)
unfolding \(*: \text{part-circlepath}-rotate-left \) \( \text{path-image-compose} \) \( \text{path-image-semicircle-Re-ge}[OF \) \( \text{assms}] \)
by auto
also have \( ?f \) \( \sphere c r = \sphere c r \)
by (rule sphere-rotate)
also have \( ?f \) \( \{s. \, \Re c \leq \Re s\} = \{s. \, \Re c \geq \Re s\} \)
by (auto simp: image-iff intro!: \( \exists x \in [\of\ - 2 * c - x \ for \ x] \))
finally show \( \?thesis \).
qed

**lemma** path-image-semicircle-Im-ge:
assumes \( r \geq 0 \)
shows \( \text{path-image} \ (\text{part-circlepath} \ c \ r \ 0 \ \pi) = \sphere c r \cap \{s. \, \Im s \geq \Im c\} \)
proof
let \( ?f = (\lambda z. \, c + cîs (\pi/2) * (z - c)) \)
have : part-circlepath c r 0 pi = part-circlepath c r (pi / 2 + (−pi/2)) (pi / 2 + pi/2)
  by simp
have path-image (part-circlepath c r 0 pi) =
  ?f ' sphere c r ∩ ?f ' {s. Re c ≤ Re s}
unfolding * part-circlepath-rotate-left path-image-compose path-image-semicircle-Re-ge[OF assms]
  by auto
also have ?f ' sphere c r = sphere c r
  by (rule sphere-rotate)
also have ?f ' {s. Re c ≤ Re s} = {s. Im c ≤ Im s}
  by (auto simp: image-iff intro: exI[of - c − i * (x - c) for x])
finally show ?thesis .
qed

lemma path-image-semicircle-Im-le:
assumes r ≥ 0
shows path-image (part-circlepath c r (2 * pi)) = sphere c r ∩ {s. Im s ≤ Im c}
proof –
  let ?f = (λz. c + cis (3*pi/2) * (z - c))
  have : part-circlepath c r (2*pi) = part-circlepath c r (3*pi/2 + (−pi/2)) (3*pi/2 + pi/2)
    by simp
  have path-image (part-circlepath c r (2*pi)) =
    ?f ' sphere c r ∩ ?f ' {s. Re c ≤ Re s}
  unfolding * part-circlepath-rotate-left path-image-compose path-image-semicircle-Re-ge[OF assms]
    by auto
also have ?f ' sphere c r = sphere c r
  by (rule sphere-rotate)
also have cis (3 * pi / 2) = −i
  using cis-mult[of pi pi / 2] by simp
hence ?f ' {s. Re c ≤ Re s} = {s. Im c ≥ Im s}
  by (auto simp: image-iff intro: exI[of - c + i * (x - c) for x])
finally show ?thesis .
qed

lemma eval-fds-logderiv-zeta-real:
assumes x > (1 :: real)
s shows (λp. ln (real p) / (p powr x − 1)) abs-summable-on {p. prime p} (is ?th1)
  and deriv zeta (of-real x) / zeta (of-real x) =
    - of-real (∑ₐp | prime p. ln (real p) / (p powr x − 1)) (is ?th2)
proof –
  have (λp. Re (of-real (ln (real p))) / (of-nat p powr of-real x − 1)))
    abs-summable-on {p. prime p} using assms
  by (intro abs-summable-Re eval-fds-logderiv-zeta) auto
also have ?this ⬛?th1

34
by (intro abs-summable-on-cong) (auto simp: powr-Reals-eq)
finally show ?th1 .
show ?th2 using assms
  by (subst eval-fds-logderiv-zeta) (auto simp: infsetsum-of-real [symmetric] powr-Reals-eq)
qed

lemma
  fixes a b c d :: real
  assumes ab: \( d \times a + b \geq 1 \) and c: \( c < -1 \) and d: \( d > 0 \)
  defines C \( \equiv - \left( (\ln (d \times a + b) - 1) / (c + 1) \right) \times (d \times a + b) \powr (c + 1) / (d \times (c + 1)) \)
  shows set-integrable-ln-powr-at-top:
    \( (\lambda x. (\ln (d \times x + b) \times ((d \times x + b) \powr c))) \) absolutely-integrable-on \{a<..\} (is ?th1)
  and set-lebesgue-integral-ln-powr-at-top:
    \( (\int x\in\{a<..\}. (\ln (d \times x + b) \times ((d \times x + b) \powr c)) \, dlborel) = C \) (is ?th2)
  and ln-pour-has-integral-at-top:
    \( ((\lambda x. \ln (d \times x + b) \times ((d \times x + b) \powr c)) \) has-integral C \{a<..\} (is ?th3)
proof
  define f where f = (\lambda x. \ln (d \times x + b) \times ((d \times x + b) \powr c))
  define F where F = (\lambda x. \ln (d \times x + b) - 1 / (c + 1)) \times (d \times x + b) \powr (c + 1) / (d \times (c + 1))

  have*: (F has-field-derivative f x) (at x) isCont f x f x \geq 0 \text{ if } x > a \text{ for } x
  proof
    have 1 \leq d \times a + b by fact
    also have \ldots < d \times x + b using that assms
      by (intro add-strict-right-mono mult-strict-left-mono)
    finally have gt-1: d \times x + b > 1 .
    show (F has-field-derivative f x) (at x) isCont f x using ab c d gt-1
      by (auto simp: F-def f-def divide-simps intro!: derivative-eq-intros continuous-intros)
        (auto simp: algebra-simps powr-add)?
    show f x \geq 0 using gt-1 by (auto simp: f-def)
  qed

  have limits: ((F \circ real-of-ereal) ----> F a) (at-right (ereal a))
    ((F \circ real-of-ereal) ----> 0) (at-left \infty)
  using c ab d unfolding ereal-tendsto-simps1 F-def by (real-asym; simp add: field-simps)+

  have 1: set-integrable lborel (einterval a \infty) f using ab c limits
    by (intro interval-integral-FTC-nonneg) (auto intro!: * AE-I2)
  thus 2: f absolutely-integrable-on \{a<..\}
    by (auto simp: set-integrable-def integrable-completion)
  have (LBINT x=ereal a..\infty. f x) = 0 - F a using ab c limits
    by (intro interval-integral-FTC-nonneg) (auto intro!: *)
  thus 3: ?th2
    by (simp add: interval-integral-to-infinity-eq F-def f-def C-def)

35
show \( \text{th3} \)

using set-borel-integral-eq-integral[OF 1] \( \beta \) by (simp add: has-integral-iff f-def C-def)

qed

lemma ln-fact-conv-sum-upto: \( \ln (\text{fact } n) = \sum\text{upto} \ln n \)

by (induction \( n \)) (auto simp: sum-upto-plus1 add.commute[of 1] ln-mult)

lemma real-of-nat-div: \( \text{real (a div b)} = \text{real-of-int} \lfloor \text{real a} / \text{real b} \rfloor \)

by (simp add: floor-divide-of-nat-eq)

lemma measurable-sum-upto:

fixes \( f \) :: \( 'a \Rightarrow \text{nat} \Rightarrow \text{real} \)

assumes measurable: \( \forall y. (\lambda t. f t y) \in M \rightarrow_M \text{borel} \)

assumes measurable: \( x \in M \rightarrow_M \text{borel} \)

shows \( (\lambda t. \sum\text{upto} (f t) (x t)) \in M \rightarrow_M \text{borel} \)

proof

have meas: \( (\lambda t. \text{set-lebesgue-integral lborel} \{y. y \geq 0 \land y - \text{real} \lfloor x t \rfloor \leq 0\}) (\lambda y. f t (\text{nat } \lceil y \rceil)) \)

\( \in M \rightarrow_M \text{borel} \) unfolding set-lebesgue-integral-def

by measurable

also have \( \forall f = (\lambda t. \sum\text{upto} (f t) (x t)) \)

proof

fix \( t :: 'a \)

show \( \forall t = \sum\text{upto} (f t) (x t) \)

proof (cases \( x t < 1 \))

case True

hence \( \{y. y \geq 0 \land y - \text{real} \lfloor x t \rfloor \leq 0\} = \{0\} \) by auto

thus \( \forall \text{thesis using True} \)

by (simp add: set-integral-at-point sum-upto-altdef)

next

case False

define \( n \) where \( n = \text{nat } \lfloor x t \rfloor \)

from False have \( n > 0 \) by (auto simp: n-def)

have \( *: (\lambda x. f t (\text{nat } \lfloor x t \rfloor)) \text{ has-integral } (f t) \{0<..\} \{\text{real } 0..\text{real } n\} \)

using \( \langle n > 0 \rangle \) by (intro nat-sum-has-integral-ceiling) auto

have \( **: (\lambda x. f t (\text{nat } \lfloor x t \rfloor)) \text{ absolutely-integrable-on } \{\text{real } 0..\text{real } n\} \)

proof (rule absolutely-integrable-absolutely-integrable-ubound)

show \( \langle \lceil x t \rceil \rangle \text{ absolutely-integrable-on } \{\text{real } 0..\text{real } n\} \)

using \( \langle n > 0 \rangle \) by (subst absolutely-integrable-on-iff-nonneg)

(auto simp: Max-ge-iff intro exI[of \( - f t 0 \)])

show \( \langle x t \rangle \text{ integrable-on } \{\text{real } 0..\text{real } n\} \)

using \( * \) by (simp add: has-integral-iff)

next
fix $y :: \text{real}$ assume $y : y \in \{\text{real 0..real n}\}$
have $ft (\text{nat } [y]) \leq |ft (\text{nat } [y])|$
  by simp
also have \ldots \leq (\text{MAX } n\in\{0..n\}.|ft n|)$
  using $y$ by (intro Max.coboundedI) auto
finally show $ft (\text{nat } [y]) \leq (\text{MAX } n\in\{0..n\}.|ft n|)$ .
qed
have $\sum (ft) \{0<..n\} = (\int x\in\{\text{real 0..real n}\}. ft (\text{nat } [x]) \, \text{lebesgue})$
  using has-integral-set-lebesgue[OF **] * by (simp add: has-integral-iff)
also have \ldots = $(\int x\in\{\text{real 0..real n}\}. ft (\text{nat } [x]) \, \text{lborel})$
  unfolding set-lebesgue-integral-def by (subst integral-completion) auto
also have $\{\text{real 0..real n}\} = \{y. \theta \leq y \land y - (\text{real } [x t]) \leq 0\}$
  by (auto simp: n-def)
also have $\sum (ft) \{0<..n\} = \text{sum-upto } (ft) (x t)$
  by (simp add: sum-upto-altdef n-def)
finally show \?thesis ..
qed
end

2 Ingham’s Tauberian Theorem

theory Newman-Ingham-Tauberian
imports 
  HOL-\text{Real-Asymp}\text{-Real-Asymp}
  Prime-Number-Theorem-Library
begin

In his proof of the Prime Number Theorem, Newman [6] uses a Tauberian theorem that was first proven by Ingham. Newman gives a nice and straightforward proof of this theorem based on contour integration. This section will be concerned with proving this theorem.

This Tauberian theorem is probably the part of the Newman’s proof of the Prime Number Theorem where most of the “heavy lifting” is done. Its purpose is to extend the summability of a Dirichlet series with bounded coefficients from the region $\Re(s) > 1$ to $\Re(s) \geq 1$.

In order to show it, we first require a number of auxiliary bounding lemmas.

lemma newman-ingham-aux1:
  fixes $R :: \text{real}$ and $z :: \text{complex}$
  assumes $R : R > 0$ and $z : \text{norm } z = R$
  shows $\text{norm } (1 / z + z / R^2) = 2 * |\text{Re } z| / R^2$
proof –
  from $z$ and $R$ have [simp]: $z \neq 0$ by auto
  have $1 / z + z / R^2 = (R^2 + z^2) * (1 / R^2 / z)$ using $R$
Given a function that is analytic on some vertical line segment, we can find a rectangle around that line segment on which the function is also analytic.

**Lemma** analytic-on-axis-extend:

\[
\text{by (simp add: field-simps power2-eq-square)}
\]

also have \( \text{norm} \ldots = \text{norm} \left( R^2 + z^2 \right)/R \sim 3 \)

\[
\text{by (simp add: numeral-3-eq-3 z norm-divide norm-mult power2-eq-square)}
\]

also have \( R^2 + z^2 = z \star (z + \text{cnj} z) \) using complex-norm-square[of z]

\[
\text{by (simp add: z power2-eq-square algebra-simps)}
\]

also have \( \text{norm} \ldots = 2 \star |\text{Re} z| \star R \)

\[
\text{by (subst complex-add-cnj) (simp-all add: z norm-mult)}
\]

also have \( \ldots / R \sim 3 = 2 \star |\text{Re} z| / R^2 \)

using \( R \) by (simp add: field-simps numeral-3-eq-3 power2-eq-square)

finally show \( \text{?thesis} \).

qed
fixes \( y_1 \ y_2 \ x :: \text{real} \)
defines \( S \equiv \{ z. \ Re \ z = x \land \ Im \ z \in \{y_1..y_2\} \} \)
avsumes \( y_1 \leq y_2 \)
assumes \( f \ \text{analytic-on} \ S \)
obtains \( x_1 \ x_2 :: \text{real} \) \text{where} \( x_1 < x \ x_2 > x \) \( f \ \text{analytic-on} \ \text{cbox} (\text{Complex} \ x_1 \ y_1) \) \( (\text{Complex} \ x_2 \ y_2) \)
proof -
define \( C \) \text{where} \( C = \{ \text{box} a \ b \ | \ a \ b \ z. \ f \ \text{analytic-on} \ \text{box} a \ b \land z \in \text{box} a \ b \land z \in S \} \)
have \( S = \text{cbox} (\text{Complex} \ x \ y_1) (\text{Complex} \ x \ y_2) \)
by (auto simp: \text{def} \ \text{in-cbox-complex-iff})
also have \( \text{compact} \ldots \) \text{by simp}finally have \( 1: \text{compact} \ S . \)

have \( 2: S \subseteq \bigcup C \)
proof (intro subsetI)
fix \( z \) assume \( z \in S \)
from \( \{ f \ \text{analytic-on} \ S \} \) and this obtain \( a \ b \) \text{where} \( z \in \text{box} a \ b \ f \ \text{analytic-on} \ \text{box} a \ b \)
by (blast elim: \text{analytic-onE-box})
with \( \{ z \in S \} \) \text{show} \( z \in \bigcup C \)
unfolding \text{C-def} \text{by blast}
qed

have \( 3: \text{open} \ X \) if \( X \in C \) \text{for} \( X \) using that \text{by (auto simp: \text{C-def})}
from \( \text{compactE}[\text{OF} \ 1 \ 2 \ 3] \) obtain \( T \) \text{where} \( T \subseteq C \) \text{finite} \( T \ S \subseteq \bigcup T \)
by blast
define \( x_1 \) \text{where} \( x_1 = \text{Max} \ \{ \text{insert} (x-1) ((\lambda X. x + (\text{Inf} \ (\text{Re } X) - x) / 2) \cdot T) \} \)
define \( x_2 \) \text{where} \( x_2 = \text{Min} \ \{ \text{insert} (x+1) ((\lambda X. x + (\text{Sup} \ (\text{Re } X) - x) / 2) \cdot T) \} \)

have *: \( x + (\text{Inf} \ (\text{Re } X) - x) / 2 < x \land x + (\text{Sup} \ (\text{Re } X) - x) / 2 > x \) if \( X \in T \) \text{for} \( X \)
proof -
from that and \( T \) obtain \( a \ b \ s \) \text{where (simp):} \( X = \text{box} a \ b \) \text{and} \( s \in \text{box} a \ b \ s \in S \)
by (force simp: \text{C-def})
hence \( le: \text{Re } a < \text{Re } b \text{ Im } a < \text{Im } b \) \text{by (auto simp: \text{in-box-complex-iff})}
show \( \text{thesis} \) \text{using} \( le \ s \)
unfolding \( X = \text{box} a \ b \) \text{Re-image-box[OF le] Im-image-box[OF le]}
by (auto simp: \text{S-def in-box-complex-iff})
qed
from * \( T \) have \( x_1 < x \) \text{unfolding} \( x_1\text{-def} \) \text{by (subst \text{Max-less-iff})} \text{auto}
from * \( T \) have \( x_2 > x \) \text{unfolding} \( x_2\text{-def} \) \text{by (subst \text{Min-gr-iff})} \text{auto}

have \( f \ \text{analytic-on} \ (\bigcup T) \)
using \( T \) \text{by (subst \text{analytic-on-Union}) (auto simp: \text{C-def})}
moreover have \( z \in \bigcup T \) if \( z \in \text{cbox} (\text{Complex} \ x_1 \ y_1) (\text{Complex} \ x_2 \ y_2) \) \text{for} \( z \)
proof –
  from that have Complex x (Im z) ∈ S
  by (auto simp: in-cbox-complex-iff S-def)
with T obtain X where X: X ∈ T Complex x (Im z) ∈ X
  by auto
with T obtain a b where [simp]: X = box a b by (auto simp: C-def)
from X have le: Re a < Re b Im a < Im b by (auto simp: in-box-complex-iff)
from that have Re z ≤ x2 by (simp add: in-cbox-complex-iff)
also have … ≤ x + (Sup (Re ' X) − x) / 2
  unfolding x2-def by (rule Min.coboundedI)(use T X in auto)
also have … = (x + Re b) / 2
  using le unfolding ‹X = box a b› by (simp add: field-simps)
also have … < (Re b + Re b) / 2
  using X by (intro divide-strict-right-mono add-strict-right-mono)
  (auto simp: in-box-complex-iff)
also have … = Re b by simp
finally have [simp]: Re z < Re b .

have Re a = (Re a + Re a) / 2 by simp
also have … < (x + Re a) / 2
  using X by (intro divide-strict-right-mono add-strict-right-mono)
  (auto simp: in-box-complex-iff)
also have … = x + (Inf (Re ' X) − x) / 2
  using le unfolding ‹X = box a b› by (simp add: field-simps)
also have … ≤ x1 unfolding x1-def by (rule Max.coboundedI)(use T X in auto)
also have … ≤ Re z using that by (simp add: in-cbox-complex-iff)
finally have [simp]: Re z > Re a .

from X have z ∈ X by (simp add: in-box-complex-iff)
with T X show ?thesis by blast
qed

hence cbox (Complex x1 y1) (Complex x2 y2) ⊆ ⋃ T by blast
ultimately have f analytic-on cbox (Complex x1 y1) (Complex x2 y2)
  by (rule analytic-on-subset)
with ‹x1 < x› and ‹x2 > x› and that[of x1 x2] show ?thesis by blast
qed

We will now prove the theorem. The precise setting is this: Consider a Dirichlet series
F(s) = ∑ a_n n^{-s} with bounded coefficients. Clearly, this converges to an analytic function f(s) on \{s | \Re(s) > 1\}.
If f(s) is analytic on the larger set \{s | \Re(s) ≥ 1\}, F converges to f(s) for all \Re(s) ≥ 1.
The proof follows Newman’s argument very closely, but some of the precise
bounds we use are a bit different from his. Also, like Harrison, we choose a combination of a semicircle and a rectangle as our contour, whereas Newman uses a circle with a vertical cut-off. The result of the Residue theorem is the same in both cases, but the bounding of the contributions of the different parts is somewhat different.

The reason why we picked Harrison’s contour over Newman’s is because we could not understand how his bounding of the different contributions fits to his contour, and it seems likely that this is also the reason why Harrison altered the contour in the first place.

**Lemma** Newman-Ingham-1:

- fixes $F ::\ complex fds$ and $f ::\ complex \Rightarrow complex$
- assumes $coeff-bound$: $fds-nth\ F \in O(\lambda - 1)$
- assumes $f$-analytic: $f$ analytic-on $\{s.\ Re\ s \geq 1\}$
- assumes $F$-conv-f: $\forall s.\ Re\ s > 1 \implies eval-fds\ F\ s = f\ s$
- assumes $w$: $Re\ w \geq 1$
- shows $fds$-converges $F\ w$ and $eval-fds\ F\ w = f\ w$

**Proof**

- We get a bound on our coefficients and call it $C$.
- We show convergence directly by showing that the difference between the partial sums and the limit vanishes.
- Next, we extend the analyticity of $f\ (w + z)$ to the left of the complex plane within a thin rectangle that is at least as high as the circle.
from this(3) have \((\lambda z. f (w + z))\) analytic-on \(\{ s. \Re s \in \{ x_1..0\} \land \Im s \in \{- R - 1..R + 1\} \} \)

by (rule analytic-on-subset) (insert x12, auto simp: in-cbox-iff)

with \(f\)-analytic' have \((\lambda z. f (w + z))\) analytic-on

\[ \{ \{ s. \Re s \geq 0 \} \cup \{ s. \Re s \in \{ x_1..0\} \land \Im s \in \{- R - 1..R + 1\} \} \} \]

by (subst analytic-on-Un) auto

hence \((\lambda z. f (w + z))\) analytic-on \(\{ s. \Re s > 0 \lor \Im s \in \{- R - 1..<R + 1\} \land \Re s > x_1 \} \)

by (rule analytic-on-subset) auto

with \(< x_1 < \theta \) and that[of \(- x_1\)] show ?thesis by auto

qed

— The function \(f (w + z)\) is now analytic on the open box \((- l; R + 1) + i(- R + 1; R + 1)\). We call this region \(X\).

define \(X\) where \(X = \text{box} (\text{Complex} (- l) (- R - 1)) (\text{Complex} (R + 1) (R + 1))\)

have \([simp, intro]:: open X convex X by (simp-all add: X-def open-box)\)

from \(R\) have \([simp]: 0 \in X by (auto simp: X-def in-box-iff)\)

have analytic: \((\lambda z. f (w + z))\) analytic-on \(X\)

by (rule analytic-on-subset[OF \(l(\theta)\)]) (auto simp: X-def in-box-iff)

note \(f\)-analytic'[analytic-intros] = analytic-on-compose-gen[OF - analytic, unfolded o-def]

note \(f\)-holo [holomorphic-intros] = holomorphic-on-compose-gen[OF - analytic-imp-holomorphic[OF analytic], unfolded o-def]

note \(f\)-cnt [continuous-intros] = continuous-on-compose2[OF holomorphic-on-imp-continuous-on[OF analytic-imp-holomorphic[OF analytic]]]

— We now pick a smaller closed box \(X'\) inside the big open box \(X\). This is because we need a compact set for the next step. our integration path still lies entirely within \(X'\), and since \(X'\) is compact, \(f (w + z)\) is bounded on it, so we obtain such a bound and call it \(M\).

define \(\delta\) where \(\delta = \min (1/2) (l/2)\)

from \(l\) have \(\delta: \delta > 0 \delta \leq 1/2 \delta < l \) by (auto simp: \(\delta\)-def)

define \(X'\) where \(\text{box} (\text{Complex} (- \delta) (- R)) (\text{Complex} R R)\)

have \(X' \subseteq X\) unfolding \(X'\)-def X-def using \(l(1) R \delta\)

by (intro subset-box-imp) (auto simp: Basis-complex-def)

have \([intro]: \text{compact} X' by (simp add: \(X'\)-def)\)

moreover have continuous-on \(X'\) (\(\lambda z. f (w + z)\))

using \(w \subseteq X\) by (auto intro!: continuous-intros)

ultimately obtain \(M\) where \(M: M \geq 0 \forall z \in X' \Rightarrow \text{norm} (f (w + z)) \leq M\)

using continuous-on-compact-bound by blast

— Our objective is now to show that the difference between the \(N\)-th partial sum and the limit is below a certain bound (depending on \(N\)) which tends to \(\theta\) for \(N \to \infty\). We use the following bound:

define bound where
bound = (λN::nat. (2*C/R + C/N + 3*M / (pi*R+ln N) + 3*R*M / (δ*pi * N powr δ)))

have 2 * C / R < ε using M(1) R C(1) δ(1) ε
  by (auto simp: field-simps)
— Evidently this is below ε for sufficiently large N.
hence eventually (λN::nat. bound N < ε) at-top
  using M(1) R C(1) δ(1) ε unfolding bound-def by real-asymp
— It now only remains to show that the difference is indeed less than the claimed bound.
thus eventually (λN. norm (f w - eval-fds (fds-truncate N F) w) < ε) at-top
proof eventually-elim
case (elim N)
  note N = this

— Like Harrison (and unlike Newman), our integration path Γ consists of a
circle A of radius π in the right-halfplane and a box of width δ and height 2R
on the left halfplane. The latter consists of three straight lines, which we call B1 to B3.
define A where A = part-circlepath 0 R (−pi/2) (pi/2)
define B2 where B2 = linepath (Complex (−δ) R) (Complex (−δ) (−R))
define B1 where B1 = linepath (R * i) (R * i − δ)
define B3 where B3 = linepath (−R * i − δ) (−R * i)
define Γ where Γ = A +++ B1 +++ B2 +++ B3

— We first need to show some basic facts about the geometry of our integration path.
  have [simp, intro]:
    path A path B1 path B3 path B2
    valid-path A valid-path B1 valid-path B3 valid-path B2
    arc A arc B1 arc B3 arc B2
    pathfinish A = −i * R pathstart A = i * R
    pathstart B1 = i * R pathfinish B1 = R * i − δ
    pathstart B3 = −R * i − δ pathfinish B3 = −i * R
    pathstart B2 = R * i − δ pathfinish B2 = −R * i − δ using R δ
    by (simp-all add: A-def B1-def B3-def exp-eq-polar B2-def Complex-eq
      arc-part-circlepath)
  hence [simp, intro]: valid-path Γ
    by (simp add: Γ-def A-def B1-def B3-def B2-def exp-eq-polar Complex-eq)
  hence [simp, intro]: path Γ using valid-path-imp-path by blast
  have [simp]: pathfinish Γ = pathstart Γ by (simp add: Γ-def exp-eq-polar)

  have image-B2: path-image B2 = {s. Re s = −δ ∧ Im s ∈ {−R..R}}
    using R by (auto simp: closed-segment-same-Re closed-segment-eq-real-ivl B2-def)
  have image-B1: path-image B1 = {s. Re s ∈ {−δ..0} ∧ Im s = R}
    and image-B3: path-image B3 = {s. Re s ∈ {−δ..0} ∧ Im s = −R}
    using δ by (auto simp: B1-def B3-def closed-segment-same-Im closed-segment-eq-real-ivl)
have image-A: path-image A = \{s. \text{Re } s \geq 0 \land \text{norm } s = R\}
unfolding A-def using R by (subst path-image-semicircle-Re-ge) auto
also have z ∈ ... → z ∈ X’ − {0} for z
using complex-Re-le-cmod[of z] abs-Im-le-cmod[of z] δ R
by (auto simp: X'-def in-cbox-complex-iff)
hence \{s. \text{Re } s \geq 0 \land \text{norm } s = R\} ⊆ X’ − {0} by auto
finally have path-image B2 ⊆ X’ − {0} path-image A ⊆ X’ − {0}
by (auto simp: X'-def in-cbox-complex-iff image-B2 image-B1 image-B3)

note path-images = this 'X' ⊆ X'

— Γ is a simple path, which, combined with its simple geometric shape, makes reasoning about its winding numbers trivial.

from R have simple-path A unfolding A-def
by (subst simple-path-part-circlepath) auto
have simple-path Γ unfolding Γ-def
proof (intro simple-path-join-loop subsetI arc-join, goal-cases)
fix z assume: z ∈ path-image A ∩ path-image (B1 ++ B2 ++ B3)
with image-A have Re z ≥ 0 norm z = R by auto
with z R δ show: z ∈ \{pathstart A, pathstart (B1 ++ B2 ++ B3)\}
by (auto simp: path-image-join image-B1 image-B2 image-B3 complex-eq-iff)
qed (insert R, auto simp: image-B1 image-B3 path-image-join image-B2 complex-eq-iff)

— We define the integrands in the same fashion as Newman:
define g where g = (λz::complex. f (w + z) * N powr z * (1 / z + z / R^2))
define S where S = eval-fds (fds-truncate N F)
define g-S where g-S = (λz::complex. S (w + z) * N powr z * (1 / z + z / R^2))
define g-rem where g-rem = (λz::complex. rem (w + z) * N powr z * (1 / z + z / R^2))

have g-holo: g holomorphic-on X − {0} unfolding g-def
by (auto intro!: holomorphic-intros analytic-imp-holomorphic[OF analytic])

have rem-altdef: rem z = eval-fds (fds-remainder N F) z if Re z > 1 for z
proof —
have abscissa: abs-cone-abscissa F ≤ 1
using assms by (intro bounded-coeffs-imp-abs-cone-abscissa-le-1)
(subst all add: natfun-bigo-iff-Bseq)
from assms and that have: f z = eval-fds F z by auto
also have F = fds-truncate N F + fds-remainder N F
by (rule fds-truncate-plus-remainder [symmetric])
also from that have: eval-fds ... z = S z + eval-fds (fds-remainder N F) z
unfolding S-def
by (auto unfolding fds-abs-converges-imp-converges)

44
finally show \( ? \text{thesis} \) by (simp add: rem-def)
qed

— We now come to the first application of the residue theorem along the path \( \Gamma \):

\[
\begin{align*}
\text{have } & \oint[\Gamma] g = 2 \pi i \ast \text{winding-number } \Gamma \ast \text{residue } g \ast 0 \\
\text{proof } & \text{(subst Residue-theorem)} \\
& \text{show } g \text{ holomorphic-on } X \text{ – \{} 0 \text{ \}} \text{ by fact} \\
& \text{show path-image } \Gamma \subseteq X \text{ – \{} 0 \text{ \}} \text{ using path-images} \\
& \text{by (auto simp: } \Gamma \text{-def path-image-join)} \\
& \text{thus } \forall z. \ z \notin X \rightarrow \text{winding-number } \Gamma z = 0 \\
& \text{by (auto intro: simply-connected-imp-winding-number-zero[of } X] \\
& \text{convex-imp-simply-connected)} \\
\end{align*}
\]

\[\text{qed (insert path-images, auto intro: convex-connected)}\]
\[\text{also have winding-number } \Gamma \ast 0 = 1\]
\[\text{proof } \text{(rule simple-closed-path-winding-number-pos)}\]
\[\text{from } R \delta \text{ have } \forall g \in \{A,B1,B2,B3\}. \ \text{Re (winding-number } g \ast 0) > 0\]
\[\text{unfolding A-def B1-def B2-def B3-def} \\
& \text{by (auto intro!: winding-number-linepath-pos-lt winding-number-part-circlepath-pos-less)} \\
& \text{hence valid-path } \Gamma \ast 0 \notin \text{ path-image } \Gamma \land \text{Re (winding-number } \Gamma \ast 0) > 0\]
\[\text{unfolding } \Gamma \text{-def using path-images(1-4) by (intro winding-number-join-pos-combined')}\]
\[\text{auto}\]
\[\text{thus } \text{Re (winding-number } \Gamma \ast 0) > 0 \text{ by simp}\]
\[\text{qed (insert path-images (simple-path } \Gamma), \text{ auto simp: } \Gamma \text{-def path-image-join)}\]
\[\text{also have residue } g \ast 0 = f w\]
\[\text{proof }\]
\[\text{have } g = (\lambda z :: \text{complex}. \ f (w + z) \ast N \text{ powr } z \ast (1 + z^2 / R^2) / z)\]
\[\text{by (auto simp: } g\text{-def divide-simps fun-eq-iff power2-eq-square simp del: div-mult-self3 div-mult-self4 div-mult-self2 div-mult-self1}\]
\[\text{moreover from } N \text{ have residue } \ldots = f w\]
\[\text{by (subst residue-simple[of } X]\}
\[\text{(auto intro!: holomorphic-intros analytic-imp-holomorphic[of analytic]}\]
\[\text{ultimately show } ? \text{thesis by (simp only:)}\]
\[\text{qed}\]
\[\text{finally have } 2 \ast \pi i \ast i \ast f w = \oint[\Gamma] g \text{ by simp}\]
\[\text{also have } \ldots = \oint[A] g + \oint[B2] g + \oint[B1] g + \oint[B3] g \text{ unfolding } \Gamma \text{-def}\]
\[\text{by (subst contour-integral-join, (insert path-images, auto intro!: contour-integral-join contour-integrable-holomorphic-simple g-holo)[4])}\]
\[\text{finally have integral1: } 2 \ast \pi i \ast i \ast f w = \oint[A] g + \oint[B2] g + \oint[B1] g + \oint[B3] g .\]

— Next, we apply the residue theorem along a circle of radius \( R \) to another integrand that is related to the partial sum:

\[
\begin{align*}
\text{have } & \oint[\text{circlepath } 0 R] g-S = 2 \ast \pi i \ast \text{residue } g-S \ast 0 \\
\text{proof } & \text{(subst Residue-theorem)} \\
& \text{show } g-S \text{ holomorphic-on } UNIV \text{ – } \{0\} \\
& \text{by (auto simp: g-S-def S-def intro!: holomorphic-intros)}
\end{align*}
\]
\[
\text{qed (insert } R, \text{ auto simp: winding-number-circlepath-centre)}
\]

also have residue \( g-S \ 0 = S \ w \)

proof –

have \( g-S = (\lambda z :: \text{complex}. \ S (w + z) * N \ \text{powr} \ z * (1 + z^2 / R^2) / z) \)

by (auto simp: g-S-def divide-simp[s] fun- \text{eq-iff} power2- \text{eq-square})


moreover from \( N \) have residue \( ... 0 = S \ w \)

by (subst residue-simple[of \( X \)][)]

(auto intro!: holomorphic-intros simp: S-def)

ultimately show \( \text{thesis by (simp only:)} \)

qed

finally have \( 2 * \pi i * S \ w = \int \{ \text{circlepath } 0 R \} \ g-S \ .. \)

— We split this integral into integrals along two semicircles in the left and right half-plane, respectively:

also have \( ... = \int \{ \text{part-circlepath } 0 R (-\pi/2) (3\pi/2) \} \ g-S \)

proof (rule Cauchy-theorem-homotopic-loops)

show homotopic-loops \( (-\{0\}) \) (circlepath 0 \( R \))

(part-circlepath 0 \( R (-\pi/2) (3\pi/2) \)) unfolding circlepath-def

using \( R \)

by (intro homotopic-loops-part-circlepath[where \( k = 1 \)][)] auto

qed (auto simp: g-S-def S-def intro!: holomorphic-intros)

also have \( ... = \int \{ A + + + -A \} \ g-S \)

proof (rule Cauchy-theorem-homotopic-paths)

have \( *: -A = \text{part-circlepath } 0 R (\pi/2) (3\pi/2) \) unfolding A-def

by (intro part-circlepath-mirror[where \( k = 0 \)][)] auto

from \( R \) show homotopic-paths \( (-\{0\}) \) (part-circlepath 0 \( R (-\pi/2) (3\pi/2) \))

\( (A + + + -A) \)

unfolding \( * \) unfolding A-def

by (intro homotopic-paths-part-circlepath) (auto dest!: in-path-image-part-circlepath)

qed (auto simp: g-S-def S-def A-def exp- \text{eq-polar} intro!: holomorphic-intros)

also have \( ... = \int \{ A \} \ g-S + \int \{ -A \} \ g-S \) using \( R \)

by (intro contour-integral-join contour-integrable-holomorphic-simple[of \( -\{0\} \)])

(auto simp: A-def g-S-def S-def path-image-mirror dest!: in-path-image-part-circlepath

intro!: holomorphic-intros)

also have \( \int \{ -A \} \ g-S = -\int \{ A \} \ (\lambda x. \ g-S (-x)) \)

by (simp add: A-def contour-integral-mirror contour-integral-\text{neg})

finally have integral2: \( 2 * \pi i * S \ w = \int \{ A \} \ g-S - \int \{ A \} \ (\lambda x. \ g-S (-x)) \)

by simp

— Next, we show a small bounding lemma that we will need for the final estimate:

have circle-bound: \( \text{norm} (1 / z + z / R^2) \leq 2 / R \) if [simp]: \( \text{norm} z = R \)

for \( z :: \text{complex} \)

proof –

have \( \text{norm} (1 / z + z / R^2) \leq 1 / R + 1 / R \)

by (intro order.trans[OF norm-triangle-\text{ineq}] add-mono)

(insert \( R, \) simp-all add: norm-divide norm-mult power2- \text{eq-square})

46
thus \( ?\)thesis by simp

qed

— The next bound differs somewhat from Newman’s, but it works just as well. Its purpose is to bound the contribution of the two short horizontal line segments.

have \( B12\text{-bound} : \text{norm} \ (\text{integral} \ \{-\delta..0\} \ (\lambda x. \ g \ (x + R' * i))) \leq 3 \ast M / R / \ln N \)
(is \( ?I \leq - \) if \( |R'| = R \) for \( R' \))

proof –

have \( ?I \leq \text{integral} \ \{-\delta..0\} \ (\lambda x. \ 3 \ast M / R \ast N \text{ powr} \ x) \)
proof (rule integral-norm-bound-integral)

fix \( x \) assume \( x \in \{-\delta..0\} \)
define \( z \) where \( z = x + i \ast R' \)
from \( R \) that have [simp]: \( z \neq 0 \ Re \ z = x \) \( \Im z = R' \)
by (auto simp: z-def complex-eq-iff)

from \( x \ R \) that have \( z \in X' \) by (auto simp: z-def X'-def in-cbox-complex-iff)
from \( x \ R \) that have \( \text{norm} \ z \leq \delta + R \)
by (intro order.trans[OF cmod-le add-mono]) auto

hence \( \text{norm} \ (1 / z + z / R^2) \leq 1 / R + (\delta / R + 1) / R \)
using \( R \) that abs-Im-le-cmod[of \( z \)]
by (intro order.trans[OF norm-triangle-ineq add-mono])
(auto simp: norm-divide norm-mult powr2-eq-square field-simps)
also have \( \delta / R \leq 1 \) using \( \delta R \) by auto
finally have \( \text{norm} \ (1 / z + z / R^2) \leq 3 / R \)
using \( R \) by (simp add: divide-right-mono)
hence \( \text{norm} \ (g \ z) \leq M \ast N \text{ powr} \ x \ast (3 / R) \)
unfolding \( g\text{-def} \) norm-mult using \( \langle M \geq 0, z \in X' \rangle \)
by (intro mult-mono mult-nonneg-nonneg M) (auto simp: norm-powr-real-powr)
thus \( \text{norm} \ (g \ (x + R' * i)) \leq 3 \ast M / R \ast N \text{ powr} \ x \) by (simp add: mult-ac z-def)

qed (insert \( N \ R \ l \) that \( \delta \), auto intro!: integrable-continuous-real-continuous-intros

simp: g-def X-def complex-eq-iff in-box-complex-iff)
also have \( \ldots = 3 \ast M / R \ast \text{integral} \ \{-\delta..0\} \ (\lambda x. \ N \text{ powr} \ x) \) by simp
also have \( ((\lambda x. \ N \text{ powr} \ x) \ \text{has-integral} \ (N \text{ powr} \ 0 / \ln N) - N \text{ powr} (-\delta) / \ln N)) \ \{-\delta..0\} \)
using \( \delta N \)
by (intro fundamental-theorem-of-calculus)
(auto simp: has-real-derivative-iff-has-vector-derivative [symmetric]

powr-def
intro!: derivative-eq-intros)

hence \( \text{integral} \ \{-\delta..0\} \ (\lambda x. \ N \text{ powr} \ x) = 1 / \ln (\text{real} N) - \text{real} N \text{ powr} - \delta / \ln (\text{real} N) \)
using \( N \) by (simp add: has-integral-iff)
also have \( \ldots \leq 1 / \ln (\text{real} N) \) using \( N \) by simp

finally show \( ?\)thesis using \( M R \) by (simp add: mult-left-mono divide-right-mono)

qed
— We combine the two results from the residue theorem and obtain an integral representation of the difference between the partial sums and the limit:

\[
\text{have } 2 * \pi * i * (f w - s w) = \int A g - \int A g-S + \int A (\lambda x. g-S (-x)) + \int B1 g + \int B3 g + \int B2 g.
\]

unfolding ring-distrib integral1 integral2 by (simp add: algebra-simps)
also have \( \int A g - \int A g-S = \int A (\lambda x. g x - g-S x) \) using path-images
by (intro contour-integral-diff [symmetric])
(auto intro: contour-integrable-holomorphic-simple[of - X - \{0\}] holomorphic-intros
simp: g-S-def g-holo S-def)
also have \( \ldots = \int A g-rem \)
by (simp add: g-rem-def g-S-def algbera-simps rem-def)
finally have \( 2 * \pi * i * (f w - s w) = \int A g-rem + \int A (\lambda x. g-S (-x)) + \int B1 g + \int B3 g + \int B2 g . \)

— We now bound each of these integrals individually:
also have \( \text{norm} \ldots \leq 2 * C * \pi / R + 2 * C * \pi * (1 / N + 1 / R) + 3 * M / R / \ln N + 3 * M / R / \ln N + 6 * R * M * N \text{ powr} (-\delta) / \delta \)
proof (rule order.trans[OF norm-triangle-ineq add-mono]+)
have \( \int B1 g = -\int \text{reversepath} B1 \) g by (simp add: contour-integral-reversepath)
also have \( \int \text{reversepath} B1 \) g = \( \int \text{integral} \{-\delta..0\} (\lambda x. g (x + R * i)) \)
unfolding \( B1\)-def reversepath-linepath using \( \delta \)
by (subst contour-integral-linepath-same-Im) auto
also have \( \text{norm} (\ldots) = \text{norm} \ldots \) by simp
also have \( \ldots \leq 3 * M / R / \ln N \) using \( R \) by (intro B12-bound) auto
finally show \( \text{norm} (\int B1) g \leq \ldots . \)
next
have \( \int B3 g = \int \text{integral} \{-\delta..0\} (\lambda x. g (x + (-R) * i)) \) unfolding B3-def using \( \delta \)
by (subst contour-integral-linepath-same-Im) auto
also have \( \text{norm} \ldots \leq 3 * M / R / \ln N \) using \( R \) by (intro B12-bound)
auto
finally show \( \text{norm} (\int B3) g \leq \ldots . \)
next
have \( \text{norm} (\int B2) g \leq M * N \text{ powr} (-\delta) * (3 / \delta) * \text{norm} (\text{Complex} (-\delta) \ (-R) - \text{Complex} (-\delta) \ R) \) unfolding B2-def
proof (rule contour-integral-bound-linepath; (fold B2-def)?), goal-cases
\( \text{case} \ (z) \) from \( 3 \ \delta \ R \) have [simp]: \( z \neq 0 \) and Re-z: \( \text{Re} z = -\delta \) and Im-z: \( \text{Im} z \in \{-R..R\} \)
by (auto simp: closed-segment-same-Re closed-segment-eq-real-ivl)
from \( 3 \) have \( z \in X' \) using \( R \delta \) path-images by (auto simp: B2-def)
from \( 3 \ \delta \ R \) have \( \text{norm} z \leq \text{sqr}\ (\delta^2 + R^2) \) unfolding cmod-def using Re-z Im-z
by (intro real-sqrt-le-mono add-mono) (auto simp: power2-le-iff-abs-le)
from power-mono[OF this, of 2] have \( \text{norm-sqr}: \text{norm} z^2 \leq \delta^2 + R^2 \)

48
by simp

have norm (1 / z + z / R^2) ≤ (1 + (norm z)^2 / R^2) / δ

unfolding add-divide-distrib using δ R abs-Re-le-cmod[of z]
by (intro order.trans[OF norm-triangle-ineq] add-mono)
(auto simp: norm-divide norm-mult field-simps power2-eq-square Re-z)
also have \( \ldots \leq (1 + (1 + δ^2 / R^2)) / δ \) using δ R \( \in \mathbb{R} \)
unfolding X1'-def
by (intro divide-right-mono add-left-mono)
(auto simp: field-simps in-cbox-complex-iff intro!: power-mono)
also have \( δ^2 / R^2 \leq 1 \)
using δ R by (auto simp: field-simps intro!: power-mono)
finally have norm (1 / z + z / R^2) ≤ 3 / δ using δ by(simp add: divide-right-mono)
with \( \iota z \in X' \) show norm (g z) ≤ M * N powr (−δ) * (3 / δ) unfolding g-def norm-mult
by (intro mult-mono mult-nonneg-nonneg M) (auto simp: norm-powr-real-powr Re-z)

qed (insert path-images M δ, auto intro!: contour-integrable-holomorphic-simple[OF g-holo])

thus norm (λ [x] g - S (- x))) \( \leq (2 * C / (real N * R) + 2 * C / R^2) \)

proof ((rule contour-integrable-bound-part-circlepath-strong)\\(\textbf{where} \) k = \{R * i, -R*i\\})
(fold A-def\\)?), goal-cases)

case (\( \delta z \))

hence [simp]: \( \iota z \neq 0 \) and norm z = R using R
by (auto simp: A-def dest!: in-path-image-part-circlepath)

from 6 have Re z \( \neq 0 \)
using \( \langle \iota norm z = R \rangle \) by (auto simp: cmod-def abs-if complex-eq-iff split: if-splits)

with 6 have Re z \( > 0 \) using image-A by auto
have S (w - z) = (\( \sum k = 1..N.\) fds-nth F k / of-nat k powr (w - z))
by (simp add: S-def eval-fds-truncate)
also have norm \( \ldots \leq C * N powr Re z * (1 / N + 1 / Re z) \)
using \( \langle \iota Re z > 0 \rangle \) w N by (intro newman-ingham-aux2 C) auto
finally have norm (S (w - z)) \( \leq \ldots \).

hence norm (g - S (-z)) \leq
\( (C * N powr (Re z) * (1 / N + 1 / Re z)) * N powr (−Re z) * (2 \)

\( \ast Re z / R^2 \)

unfolding g-def norm-mult
using newman-ingham-aux1[OF \( \langle \iota norm z = R \rangle \) \( \langle \iota Re z > 0 \rangle \) \( \langle \iota C \geq 1 \rangle \) R]
by (intro mult-mono mult-nonneg-nonneg circle-bound)
(auto simp: norm-powr-real-powr norm-uminus-minus)
also have \( \ldots = 2 * C * (Re z / N + 1) / R^2 \) using R N \( \langle \iota Re z > 0 \rangle \)
by (simp add: powr-minus algebra-simps)
also have \ldots \leq 2 \cdot C / (N \cdot R) + 2 \cdot C / R^2 unfolding add-divide-distrib
ring-distribbs
using R N abs-Re-le-cmod[of z] \langle norm z = R \cdot \langle Re z > 0 \rangle \cdot \langle C \geq 1 \rangle
by (intro add-mono) (auto simp: power2-eq-square field-simps mult-mono)
finally show \langle case \\[ qed \langle insert R N image-A C, auto intro: contour-integrable-holomorphic-simple[of -\{0\}] \rangle holomorphic-intros simp: g-S-def S-def \rangle
also have \ldots = 2 \cdot C \cdot \pi \cdot (1 / N + 1 / R) using R N
by (simp add: power2-eq-square field-simps)
finally show norm (\langle \int[A] (\lambda x. g-S (- z)) \rangle \leq \ldots .
next
have norm (\langle \int[A] g-rem \rangle \leq (2 \cdot C / R^2) \cdot R \cdot ((\pi / 2) - (-\pi / 2))
unfolding A-def
proof ((rule contour-integral-bound-part-circlepath-strong)[where k = \{R \cdot i, -Re+i\}];
(fold A-def)?, goal-cases)
case (6 z)
  hence (simp); z \neq 0 and norm z = R using R
  by (auto simp: A-def dest!: in-path-image-part-circlepath)
  from 6 have Re z \neq 0
  using \langle norm z = R \rangle by (auto simp: cmod-def abs-if complex-eq-iff split:
  if-splits)
  with 6 have Re z > 0 using image-A by auto
have summable: summable (\langle \lambda n. C \cdot (1 / (Suc n + N)) powr (Re w + Re z) \rangle)
  using summable-hurwitz-zeta-real[of Re w + Re z Suc N] \langle Re z > 0 \rangle w
  unfolding powr-minus by (intro summable-mult) (auto simp: field-simps)
  have rem (w + z) = (\sum n. fds-nth F (Suc n + N) / (Suc n + N) powr (w + z))
  using \langle Re z > 0 \rangle w by (simp add: rem-altdef eval-fds-remainder)
  also have norm \ldots \leq (\sum n. C / (Suc n + N) powr Re (w + z)) using
  summable
  by (intro norm-suminf-le)
  (auto simp: norm-divide norm-powr-powr intro!: divide-right-mono)
\langle C \rangle
also have \ldots = (\sum n. C \cdot (Suc n + N) powr -Re (w + z))
unfolding powr-minus by (simp add: field-simps)
also have \ldots = C \cdot (\sum n. (Suc n + N) powr -Re (w + z))
unfolding powr-minus by (simp add: field-simps)
also have (\sum n. (Suc n + N) powr -Re (w + z)) \leq
  N powr (1 - Re (w + z)) / (Re (w + z) - 1)
using \langle Re z > 0 \rangle w N hurwitz-zeta-real-bound-aux[of N Re (w + z)]
by (auto simp: add-ac)
also have \ldots \leq N powr -Re z / Re z
using w N \langle Re z > 0 \rangle by (intro frac-le powr-mono) auto
Finally have \( \text{norm} \ (\text{rem} \ (w + z)) \leq C / (\text{Re} \ z * N \ \text{powr} \ \text{Re} \ z) \)

using \( C \) by (simp add: mult-left-mono mult-right-mono powr-minus field-simps)

Hence \( \text{norm} \ (g \cdot \text{rem} \ z) \leq (C / (\text{Re} \ z * N \ \text{powr} \ \text{Re} \ z)) * N \ \text{powr} \ (\text{Re} \ z) * (2 * \text{Re} \ z / R^2) \)

unfolding \( g \cdot \text{rem-def} \) norm-mult

using newman-ingham-aux1 [OF - \langle \text{norm} \ z = R \rangle R \langle \text{Re} \ z > 0 \rangle C]

by (intro mult-mono mult-nonneg-nonneg circle-bound)

(auto simp: norm-powr-real-powr norm-uminus-minus)

Also have \( \ldots = 2 * C / R^2 \) using \( R \cdot N \langle \text{Re} \ z > 0 \rangle \)

by (simp add: powr-minus field-simps)

finally show \?case .

Next

show \( g \cdot \text{rem} \ \text{contour-integrable-on} \ A \) using path-images

by (auto simp: g-rem-def rem-def S-def intro!: contour-integrable-holomorphic-simple[of - X-\{0\}]

holomorphic-intros)

qed (insert \( R \cdot N \cdot C \), auto)

also have \( (2 * C / R^2) * R * ((\pi / 2) - (-\pi / 2)) = 2 * C * \pi / R \)

using \( R \) by (simp add: powr2-eq-square field-simps)

finally show \( \text{norm} \ (\lambda [A] g \cdot \text{rem}) \leq \ldots . \)

qed

also have \( \ldots = 4 * C * \pi / R + 2 * C * \pi / N + 6 * M / R / ln N + 6 * R * M * N \)

powr - \( \delta / \delta \)

by (simp add: algebra-simps)

also have \( \ldots = 2 * \pi / (2 * C / R + C / N + 3 * M / (\pi * R * \ln N) + 3 * R * M \)

/ \( (\delta * \pi * N \ \text{powr} \ \delta)) \)

by (simp add: field-simps powr-minus )

also have \( \text{norm} \ (2 * \pi * i * (f \ w - S \ w)) = 2 * \pi * \text{norm} \ (f \ w - S \ w) \)

by (simp add: norm-mult)

finally have \( \text{norm} \ (f \ w - S \ w) \leq \text{bound} \ N \) by (simp add: bound-def)

also have \( \text{bound} \ N < \varepsilon \) by fact

finally show \( \text{norm} \ (f \ w - S \ w) < \varepsilon . \)

qed

qed

thus \( \text{fds-converges} \ F \ w \)

by (auto simp: fds-converges-altdef intro: convergentI)

thus \( \text{eval-fds} \ F \ w = f \ w \)

using \( \langle \lambda N. \ \text{eval-fds} \ (\text{fds-truncate} \ N \ F) \ w \rangle \longrightarrow f \ w \)

by (intro tendsto-unique[of - \text{tendsto-vals-fds-truncate}]) auto

qed

The theorem generalises in a trivial way; we can replace the requirement
that the coefficients of \( f(s) \) be \( O(1) \) by \( O(n^\sigma - 1) \) for some \( \sigma \in \mathbb{R} \), then \( f(s) \) converges for \( \Re(s) > \sigma \). If it can be analytically continued to \( \Re(s) \geq \sigma \), it is also convergent there.

Theorem Newman-Ingham:

fixes \( F :: \text{complex} \) fds and \( f :: \text{complex} \Rightarrow \text{complex} \)

assumes \( \text{coeff-bound}: \ \text{fds-nth} \ F \in O(\lambda n. n \ \text{powr} \ \text{of-real} \ (\sigma - 1)) \)
assumes $f$-analytic: $f$ analytic-on \( \{ s \mid \text{Re } s \geq \sigma \} \)

assumes $F$-conv-f: $\forall s. \text{Re } s > \sigma \iff \text{eval-fds } F \ s = f \ s$

assumes $w$: $\text{Re } w \geq \sigma$

shows $\text{fds-converges } F \ w$ and $\text{eval-fds } F \ w = f \ w$

proof –

define $F'$ where $F' = \text{fds-shift } (-\text{of-real } (\sigma - 1)) F$

define $f'$ where $f' = f \circ (\lambda s. s + \text{of-real } (\sigma - 1))$

have $\text{fds-nth } F' = (\lambda n. \text{fds-nth } F \ n * \text{of-nat } n \text{ powr } -\text{of-real}(\sigma - 1))$ by (auto simp: fun-eq-iff $F'$-def)

also have $\ldots \in O(\lambda n. \text{of-nat } n \text{ powr } -\text{of-real}(\sigma - 1))$ by (intro landau-o.big.mult-right assms)

also have $(\lambda n. \text{of-nat } n \text{ powr } -\text{of-real}(\sigma - 1)) \in \Theta(\lambda n. 1)$ by (intro bigthetaI-cong eventually-mono [OF eventually-gt-at-top [of 0]])

(auto simp: powr-minus powr-diff)

finally have $\text{bigo: } \text{fds-nth } F' \in O(\lambda n. 1)$.

from $f$-analytic have analytic: $f'$ analytic-on \( \{ s \mid \text{Re } s \geq 1 \} \) unfolding $f'$-def by (intro analytic-on-compose-gen [OF - $f$-analytic]) (auto intro: analytic-intros)

have $F'$-f: $\text{eval-fds } F' \ s = f' \ s$ if $\text{Re } s > 1$ for $s$

using assms that by (auto simp: $F'$-def $f'$-def algebra-simps)

have $w'$: $1 \leq \text{Re } (w - \text{of-real } (\sigma - 1))$

using $w$ by simp

have $1$: $\text{fds-converges } F' (w - \text{of-real } (\sigma - 1))$

using bigo analytic $F'$-f $w'$ by (rule Newman-Ingham-1)

thus $\text{fds-converges } F \ w$ by (auto simp: $F'$-def)

have $2$: $\text{eval-fds } F' (w - \text{of-real } (\sigma - 1)) = f' (w - \text{of-real } (\sigma - 1))$

using bigo analytic $F'$-f $w'$ by (rule Newman-Ingham-1)

thus $\text{eval-fds } F \ w = f \ w$

using assms by (simp add: $F'$-def $f'$-def)

qed

end

3 Prime-Counting Functions

theory Prime-Counting-Functions
  imports Prime-Number-Theorem-Library
begin

We will now define the basic prime-counting functions $\pi$, $\vartheta$, and $\psi$. Additionally, we shall define a function $M$ that is related to Mertens’ theorems and Newman’s proof of the Prime Number Theorem. Most of the results in
this file are not actually required to prove the Prime Number Theorem, but are still nice to have.

3.1 Definitions

definition prime-sum-upto :: (nat ⇒ 'a) ⇒ real ⇒ 'a :: semiring_1 where
   prime-sum-upto f x = (∑ p. prime p ∧ real p ≤ x. f p)

lemma prime-sum-upto-altdef1:
   prime-sum-upto f x = sum-upto (λp. ind prime p * f p) x
unfolding sum-upto-def prime-sum-upto-def
by (intro sum.mono-neutral cong-left finite-subset[OF - finite-Nats-le-real[of x]])
   (auto dest: prime-gt-1-nat simp: ind-def)

lemma prime-sum-upto-altdef2:
   prime-sum-upto f x = (∑ p. prime p ∧ p ≤ nat ⌊x⌋. f p)
unfolding sum-upto-altdef prime-sum-upto-altdef1
by (intro sum.mono-neutral cong-right) (auto simp: ind-def dest: prime-gt-1-nat)

lemma prime-sum-upto-altdef3:
   prime-sum-upto f x = (∑ p←primes-upto (nat ⌊x⌋). f p)

proof −
   have (∑ p←primes-upto (nat ⌊x⌋). f p) = (∑ p. prime p ∧ p ≤ nat ⌊x⌋. f p)
      using assms by (intro sum-list-distinct-cone-sum-set) (auto simp: set-primes-upto conj-commute)
   thus ?thesis by (simp add: prime-sum-upto-altdef2)
qed

lemma prime-sum-upto-eqI:
   assumes a ≤ b ⋀ k ∈ {nat [a]<..nat[b]} =⇒ ¬prime k
   shows prime-sum-upto f a = prime-sum-upto f b
proof −
   have *: k ≤ nat [a] if k ≤ nat [b] prime k for k
      using that assms(2)[of k] by (cases k ≤ nat [a]) auto
   from assms(1) have nat [a] ≤ nat [b] by linarith
   hence (∑ p. prime p ∧ p ≤ nat [a]. f p) = (∑ p. prime p ∧ p ≤ nat [b]. f p)
      using assms by (intro sum.mono-neutral-left) (auto dest: *)
   thus ?thesis by (simp add: prime-sum-upto-altdef2)
qed

lemma prime-sum-upto-eqI':
   assumes a' ≤ nat [a] a ≤ b nat [b] ≤ b' ⋀ k. k ∈ {a'<..<b'} =⇒ ¬prime k
   shows prime-sum-upto f a = prime-sum-upto f b
by (rule prime-sum-upto-eqI) (use assms in auto)

lemmas eval-prime-sum-upto = prime-sum-upto-altdef3[unfolded primes-upto-sieve]

lemma of-nat-prime-sum-upto: of-nat (prime-sum-upto f x) = prime-sum-upto (λp. of-nat (f p)) x
   by (simp add: prime-sum-upto-def)
lemma prime-sum-upato-mono:
assumes \( \forall n. n > 0 \implies f n \geq (0 :: \text{real}) \) \( x \leq y \)
shows \( \text{prime-sum-upato} f x \leq \text{prime-sum-upato} f y \)
using assms unfolding prime-sum-upato-altdef1 sum-upato-altdef
by (intro sum-mono2) (auto simp: le-nat-iff le-floor-iff ind-def)

lemma prime-sum-upato-nonneg:
assumes \( \forall n. n > 0 \implies f n \geq (0 :: \text{real}) \)
shows \( \text{prime-sum-upato} f x \geq 0 \)
unfolding prime-sum-upato-altdef1 sum-upato-altdef
by (intro sum-nonneg) (auto simp: ind-def assms)

lemma prime-sum-upato-eq-0:
assumes \( x < 2 \)
shows \( \text{prime-sum-upato} f x = 0 \)
proof
  from assms have \( \text{nat} \lfloor x \rfloor = 0 \lor \text{nat} \lfloor x \rfloor = 1 \) by linarith
  thus \?thesis by (auto simp: eval-prime-sum-upato)
qed

lemma measurable-prime-sum-upato [measurable]:
  fixes \( f :: 'a \Rightarrow \text{nat} \Rightarrow \text{real} \)
  assumes \( \forall y. (\lambda t. f t y) \in M \rightarrow M \text{ borel} \)
  assumes \( x \in M \rightarrow M \text{ borel} \)
  shows \( (\lambda t. \text{prime-sum-upato} (f t) (x t)) \in M \rightarrow M \text{ borel} \)
  unfolding prime-sum-upato-altdef1 by measurable

The following theorem breaks down a sum over all prime powers no greater
than fixed bound into a nicer form.

lemma sum-upato-primepows:
  fixes \( f :: \text{nat} \Rightarrow 'a :: \text{comm-monoid-add} \)
  assumes \( \forall n. \neg \text{primepow} n \implies f n = 0 \\\forall i. \text{prime} p \implies i > 0 \implies f (p ^ i) = g p i \)
  shows \( \text{sum-upato} f x = (\sum (p, i) \mid \text{prime} p \land i > 0 \land \text{real} (p ^ i) \leq x. g p i) \)
proof
  let \(?d = \text{aprimedivisor}\)
  have \( g : g (?d n) \ (\text{multiplicity} (?d n) n) = f n \) if primepow n for n using that
    by (subst assms(2) [symmetric])
    (auto simp: primepow-decompose aprimedivisor-prime-power primepow-gt-Suc-0
    intro: aprimedivisor-nat multiplicity-aprimedivisor-gt-0-nat)
  have \( \text{sum-upato} f x = (\sum n \mid \text{primepow} n \land \text{real} n \leq x. f n) \)
  unfolding sum-upato-def using assms
  by (intro sum.mono-neutral-cong-right) (auto simp: primepow-gt-0-nat)
  also have \( \ldots = (\sum (p, i) \mid \text{prime} p \land i > 0 \land \text{real} (p ^ i) \leq x. g p i) \) (is - = sum - ?S)
    by (rule sum.reindex-bij-witness[of - \lambda(p,i). p ^ i \land. (?d n, \text{multiplicity} (?d n) n)])
    (auto simp: aprimedivisor-prime-power primepow-decompose primepow-gt-Suc-0

54
Next, we define some nice optional notation for these functions.

bundle prime-counting-notation
begin

notation primes-pi (π)
notation primes-theta (ϑ)
notation primes-psi (ψ)
notation primes-M (M)

end

bundle no-prime-counting-notation
begin

no-notation primes-pi (π)
nо-notation primes-theta (ϑ)
nо-notation primes-psi (ψ)
nо-notation primes-M (M)

end

lemmas π-def = primes-pi-def
lemmas ϑ-def = primes-theta-def
lemmas ψ-def = primes-psi-def

lemmas eval-π = primes-pi-def[unfolded eval-prime-sum-upto]
lemmas eval-ϑ = primes-theta-def[unfolded eval-prime-sum-upto]
lemmas eval-M = primes-M-def[unfolded eval-prime-sum-upto]

3.2 Basic properties

The proofs in this section are mostly taken from Apostol [1].

lemma measurable-π [measurable]; π ∈ borel →_M borel
and measurable-ϑ [measurable]; ϑ ∈ borel →_M borel
and measurable-ψ [measurable]; ψ ∈ borel →_M borel
and measurable-primes-M [measurable]; M ∈ borel →_M borel
unfolding primes-M-def π-def ϑ-def ψ-def by measurable
lemma π-eq-0 [simp]: \( x < 2 \implies \pi x = 0 \)
and \( \vartheta\)-eq-0 [simp]: \( x < 2 \implies \vartheta x = 0 \)
and primes-M-eq-0 [simp]: \( x < 2 \implies \mathfrak{M} x = 0 \)
unfolding primes-pi-def primes-theta-def primes-M-def
by (rule prime-sum-upto-eq-0; simp)+

lemma π-nat-cancel [simp]: \( \pi (nat x) = \pi x \)
and \( \vartheta\)-nat-cancel [simp]: \( \vartheta (nat x) = \vartheta x \)
and primes-M-nat-cancel [simp]: \( \mathfrak{M} (nat x) = \mathfrak{M} x \)
and \( \psi\)-nat-cancel [simp]: \( \psi (nat x) = \psi x \)
and primes-M-floor-cancel [simp]: \( \mathfrak{M} (of-int \lfloor y \rfloor) = \mathfrak{M} y \)
and \( \psi\)-floor-cancel [simp]: \( \psi (of-int \lfloor y \rfloor) = \psi y \)
by (simp-all add: \( \pi\)-def \( \vartheta\)-def \( \psi\)-def primes-M-def prime-sum-upto-altdef2 sum-upto-altdef)

lemma π-nonneg [intro]: \( \pi x \geq 0 \)
and \( \vartheta\)-nonneg [intro]: \( \vartheta x \geq 0 \)
and primes-M-nonneg [intro]: \( \mathfrak{M} x \geq 0 \)
unfolding primes-pi-def primes-theta-def primes-M-def
by (rule prime-sum-upto-nonneg; simp)+

lemma π-mono [intro]: \( x \leq y \implies \pi x \leq \pi y \)
and \( \vartheta\)-mono [intro]: \( x \leq y \implies \vartheta x \leq \vartheta y \)
and primes-M-mono [intro]: \( x \leq y \implies \mathfrak{M} x \leq \mathfrak{M} y \)
unfolding primes-pi-def primes-theta-def primes-M-def
by (rule prime-sum-upto-mono; simp)+

lemma π-pos-iff: \( \pi x > 0 \iff x \geq 2 \)
proof
  assume: \( x \geq 2 \)
  show: \( \pi x > 0 \)
    by (rule less-le-trans[OF \( \pi\)-mono[OF \( x\)]]) (auto simp: eval-π)
next
  assume: \( \pi x > 0 \)
  hence \( \neg (x < 2) \) by auto
  thus: \( x \geq 2 \) by simp
qed

lemma π-pos: \( x \geq 2 \implies \pi x > 0 \)
by (simp add: π-pos-iff)

lemma \( \psi\)-eq-0 [simp]:
  assumes: \( x < 2 \)
  shows: \( \psi x = 0 \)
proof
  from \( \text{assms} \) have: \( \text{nat } x \leq 1 \) by linarith
  hence: \( \text{mangoldt } n = (0 :: real) \text{ if } n \in \{0<..\text{nat } x\} \) for \( n \)
using that by (auto simp: mangoldt-def dest!: primepow-gt-Suc-0)
thus thesis unfolding ψ-def sum-upto-altdef by (intro sum.neutral) auto

qed

lemma ψ-nonneg [intro]: ψ x ≥ 0
  unfolding ψ-def sum-upto-def by (intro sum-nonneg mangoldt-nonneg)

lemma ψ-mono: x ≤ y ⇒ ψ x ≤ ψ y
  unfolding ψ-def sum-upto-def by (intro sum-mono2 mangoldt-nonneg) auto

3.3 The n-th prime number

Next we define the n-th prime number, where counting starts from 0. In traditional mathematics, it seems that counting usually starts from 1, but it is more natural to start from 0 in HOL and the asymptotics of the function are the same.

definition nth-prime :: nat ⇒ nat where
  nth-prime n = (THE p. prime p ∧ card {q. prime q ∧ q < p} = n)

lemma finite-primes-less [intro]: finite {q::nat. prime q ∧ q < p}
  by (rule finite-subset[of - {..<p}]) auto

lemma nth-prime-unique-aux:
  fixes p p' :: nat
  assumes prime p card {q. prime q ∧ q < p} = n
  assumes prime p' card {q. prime q ∧ q < p'} = n
  shows p = p'
  using assms
  proof (induction p p' rule: linorder-wlog)
    case (le p p')
    have finite {q. prime q ∧ q < p'} by (rule finite-primes-less)
    moreover from le have {q. prime q ∧ q < p} ⊆ {q. prime q ∧ q < p'}
      by auto
    moreover from le have card {q. prime q ∧ q < p} = card {q. prime q ∧ q < p'}
      by simp
    ultimately have {q. prime q ∧ q < p} = {q. prime q ∧ q < p'}
      by (rule card-subset-eq)
    with ⟨prime p ⟩ have ¬(p < p') by blast
    with ⟨p ≤ p' ⟩ show p = p' by auto
  qed auto

lemma π-smallest-prime-beyond:
  π (real (smallest-prime-beyond m)) = π (real (m - 1)) + 1
  proof (cases m)
    case 0
    have smallest-prime-beyond 0 = 2
      by (rule smallest-prime-beyond-eq) (auto dest: prime-gt-1-nat)
with \( \theta \) show \( \text{thesis} \) by (simp add: eval-\( \pi \))

next

\text{case} (Suc \( n \))

\text{define} \( n' \ where \ n' = \text{smallest-prime-beyond} (\text{Suc} \ n) \)

\text{have} \( n < n' \)

\text{using} \( \text{smallest-prime-beyond-le} [\text{of} \ \text{Suc} \ n] \) \text{unfolding} \( n' \)-\text{def} by linarith

\text{have} \( \text{prime} \ n' \) by (simp add: \( n' \)-\text{def})

\text{have} \( n' \leq p \) if \( \text{prime} \ p \) \( p > n \) for \( p \)

\text{using} \( \text{that} \ \text{smallest-prime-beyond-smallest} [\text{of} \ p \ \text{Suc} \ n] \) by (auto simp: \( n' \)-\text{def})

\text{note} \( n' = \langle n < n' \rangle \langle \text{prime} \ n' \rangle \)

\text{have} \( \pi (\text{real} \ n') = \text{real} \ (\text{card} \ \{ p. \ \text{prime} \ p \land p \leq n' \}) \)

\text{by} (simp add: \( \pi \)-\text{def} prime-sum-upto-def)

\( \)also have \( \text{Suc} \ n \leq n' \) \text{unfolding} \( n' \)-\text{def} by (rule smallest-prime-beyond-le)

\text{hence} \( \{ p. \ \text{prime} \ p \land p \leq n' \} = \{ p. \ \text{prime} \ p \land p \leq n \} \cup \{ p. \ \text{prime} \ p \land p \in \{ n < .. n' \} \}

\text{by} auto

\text{also have} \( \text{real} \ (\text{card} \ldots) = \pi \ (\text{real} \ n) + \text{real} \ (\text{card} \ \{ p. \ \text{prime} \ p \land p \in \{ n < .. n' \} \}) \)

\text{by} (\text{subst} \ \text{card-Un-disjoint}) (\text{auto simp} \ \text{prime-sum-upto-def})

\text{also have} \( \{ p. \ \text{prime} \ p \land p \in \{ n < .. n' \} \} = \{ n' \} \)

\text{using} \( n' \) by (auto intro: antisym)

\text{finally show} \ \text{thesis} using \text{Suc by} (simp add: \( n' \)-\text{def})

\text{qed}

\text{lemma} \( \pi \)-inverse-exists: \( \exists n. \ \pi (\text{real} \ n) = \text{real} \ m \)

\text{proof} (induction \( m \))

\text{case} 0

\text{show} \ \text{case} by (intro exI [of - 0]) auto

next

\text{case} (Suc \( m \))

\text{from} \text{Suc.IH obtain} \( n \) \text{where} \( n: \pi (\text{real} \ n) = \text{real} \ m \)

\text{by} auto

\text{hence} \( \pi (\text{real} (\text{smallest-prime-beyond} (\text{Suc} \ n))) = \text{real} (\text{Suc} \ m) \)

\text{by} (\text{subst} \ \pi\text{-smallest-prime-beyond}) auto

\text{thus} \ \text{case by blast}

\text{qed}

\text{lemma} \ nth-prime-exists: \( \exists p::\text{nat.} \ \text{prime} \ p \land \text{card} \ \{ q. \ \text{prime} \ q \land q < p \} = n \)

\text{proof}

\text{from} \( \pi\text{-inverse-exists}[\text{of} \ n] \) \text{obtain} \( m \) \text{where} \( \pi (\text{real} \ m) = \text{real} \ n \) \text{by blast}

\text{hence} \text{card: card} \ \{ q. \ \text{prime} \ q \land q \leq m \} = n

\text{by} (\text{auto simp:} \ \pi\text{-def} prime-sum-upto-def)

\text{define} \( p \) \text{where} \( p = \text{smallest-prime-beyond} (\text{Suc} \ m) \)

\text{have} \( m < p \) \text{using} \( \text{smallest-prime-beyond-le} [\text{of} \ \text{Suc} \ m] \) \text{unfolding} \( p\)-\text{def} by linarith

\text{have} \( \text{prime} \ p \) by (simp add: \( p\)-\text{def})

\text{have} \( p \leq q \) if \( \text{prime} \ q \) \( q > m \) for \( q \)

\text{using} \( \text{smallest-prime-beyond-smallest} [\text{of} \ q \ \text{Suc} \ m] \) \text{by} (simp add: \( p\)-\text{def})
\[ p = \langle m < p, \text{prime} \rangle \text{ this} \]

**have** \( \{ q. \text{prime} q \land q < p \} = \{ q. \text{prime} q \land q \leq m \} \)

**proof**
- **safe**
- **fix** \( q \) **assume** \( \text{prime} q \land q < p \)
- **hence** \( \neg (q > m) \) **using** \( p(1,2) \) \( p(3)[\text{of} q] \) **by** \( \text{auto} \)
- **thus** \( q \leq m \) **by** \( \text{simp} \)

**qed** (insert \( p \), \( \text{auto} \))

**also have** \( \text{card} \ldots = n \) **by** \( \text{fact} \)

**finally show** \( \text{thesis using } \langle \text{prime} p \rangle \) **by** \( \text{blast} \)

**qed**

**lemma** nth-prime-exists1: \( \exists! p :: \text{nat}. \text{prime} p \land \text{card} \{ q. \text{prime} q \land q < p \} = n \)

**by** (intro ex-ex1I nth-prime-exists) (blast intro: nth-prime-unique-aux)

**lemma** prime-nth-prime [intro]: \( \text{prime} (\text{nth-prime} n) \)

and **card-less-nth-prime** [simp]: \( \text{card} \{ q. \text{prime} q \land q < \text{nth-prime} n \} = n \)

**using** the1’ [OF nth-prime-exists1[of n]] **by** (simp-all add: nth-prime-def)

**lemma** card-le-nth-prime [simp]: \( \text{card} \{ q. \text{prime} q \land q \leq \text{nth-prime} n \} = \text{Suc} n \)

**proof**
- **have** \( \{ q. \text{prime} q \land q \leq \text{nth-prime} n \} = \text{insert} (\text{nth-prime} n) \{ q. \text{prime} q \land q < \text{nth-prime} n \} \)
  **by** \( \text{auto} \)
- **also have** \( \text{card} \ldots = \text{Suc} n \) **by** \( \text{simp} \)
- **finally show** \( \text{thesis} \).

**qed**

**lemma** π-nth-prime [simp]: \( \pi (\text{real} (\text{nth-prime} n)) = \text{real} n + 1 \)

**by** (simp add: π-def prime-sum-upto-def)

**lemma** nth-prime-eqI:
- **assumes** \( \text{prime} p \) \( \text{card} \{ q. \text{prime} q \land q < p \} = n \)
- **shows** \( \text{nth-prime} n = p \)
- **unfolding** nth-prime-def
- **by** (rule the1-equality[OF nth-prime-exists1]) (use assms in auto)

**lemma** nth-prime-eqI’:
- **assumes** \( \text{prime} p \) \( \text{card} \{ q. \text{prime} q \land q \leq p \} = \text{Suc} n \)
- **shows** \( \text{nth-prime} n = p \)
- **proof** (rule nth-prime-eqI)
  - **have** \( \{ q. \text{prime} q \land q \leq p \} = \text{insert} p \{ q. \text{prime} q \land q < p \} \)
    **using** assms **by** \( \text{auto} \)
  - **also have** \( \text{card} \ldots = \text{Suc} (\text{card} \{ q. \text{prime} q \land q < p \}) \)
    **by** \( \text{simp} \)
  - **finally show** \( \text{card} \{ q. \text{prime} q \land q < p \} = n \) **using** assms **by** \( \text{simp} \)
  **qed** (use assms in auto)

**lemma** nth-prime-eqI’’:

59
assumes prime p π (real p) = real n + 1
shows nth-prime n = p
proof (rule nth-prime-eqI)
  have real (card {q. prime q ∧ q ≤ p}) = π (real p)
  by (simp add: π-def prime-sum-upto-def)
  also have ... = real (Suc n) by (simp add: assms)
finally show card {q. prime q ∧ q ≤ p} = Suc n
  by (simp only: of-nat-eq-iff)
qed fact+

lemma nth-prime-0 [simp]: nth-prime 0 = 2
  by (intro nth-prime-eqI) (auto dest: prime-gt-1-nat)

lemma nth-prime-Suc: nth-prime (Suc n) = smallest-prime-beyond (Suc (nth-prime n))
  by (rule nth-prime-eqI"") (simp-all add: π-smallest-prime-beyond)

lemmas nth-prime-code [code] = nth-prime-0 nth-prime-Suc

lemma strict-mono-nth-prime: strict-mono nth-prime
proof (rule strict-monoI-Suc)
  fix n :: nat
  have Suc (nth-prime n) ≤ smallest-prime-beyond (Suc (nth-prime n)) by simp
  also have ... ≤ nth-prime (Suc n) by (simp add: nth-prime-Suc)
  finally show nth-prime n < nth-prime (Suc n) by simp
qed

lemma nth-prime-le-iff [simp]: nth-prime m ≤ nth-prime n ←→ m ≤ n
  using strict-mono-le-iff[OF strict-mono-nth-prime] by blast

lemma nth-prime-less-iff [simp]: nth-prime m < nth-prime n ←→ m < n
  using strict-mono-less[OF strict-mono-nth-prime] by blast

lemma nth-prime-eq-iff [simp]: nth-prime m = nth-prime n ←→ m = n
  using strict-mono-eq[OF strict-mono-nth-prime] by blast

lemma nth-prime-ge-2: nth-prime n ≥ 2
  using nth-prime-le-iff[of 0 n] by (simp del: nth-prime-le-iff)

lemma nth-prime-lower-bound: nth-prime n ≥ Suc (Suc n)
proof –
  have n = card {q. prime q ∧ q < nth-prime n}
    by simp
  also have ... ≤ card {2..<nth-prime n}
    by (intro card-mono) (auto dest: prime-gt-1-nat)
  also have ... = nth-prime n - 2 by simp
  finally show thesis using nth-prime-ge-2[of n] by linarith
qed

60
lemma nth-prime-at-top: filterlim nth-prime at-top at-top
proof (rule filterlim-at-top-mono)
  show filterlim (\(\lambda n::\text{nat}. \ n + 2\)) at-top at-top by real-asym
qed (auto simp: nth-prime-lower-bound)

lemma \(\pi\)-at-top: filterlim \(\pi\) at-top at-top
unfolding filterlim-at-top
proof safe
  fix C :: real
  define x0 where x0 = real (nth-prime (\text{nat} \(\lceil\max 0 \ C\rceil\)))
  show eventually (\(\lambda x. \ \pi \ x \geq \ C\)) at-top
  using eventually-ge-at-top
  proof (eventually-elim)
    fix x assume x \geq x0
    have C \leq real (\text{nat} \(\lceil\max 0 \ C\rceil + 1\)) by linarith
    also have real (\text{nat} \(\lceil\max 0 \ C\rceil + 1\)) = \(\pi\) x0
    unfolding x0-def by simp
    also have \ldots \leq \(\pi\) x \(\text{by (rule \(\pi\)-mono)}\) fact
    finally show \(\pi\) x \geq \ C .
  qed
qed

An unbounded, strictly increasing sequence \(a_n\) partitions \([a_0; \infty)\) into segments of the form \([a_n; a_{n+1})\).

lemma strict-mono-sequence-partition:
  assumes strict-mono \((f :: \text{nat} \Rightarrow 'a::{\text{linorder}, no-top})\)
  assumes x \geq f 0
  assumes filterlim f at-top at-top
  shows \(\exists k. \ x \in \{f k..<f (Suc \ k)\}\)
proof (-
  define k where k = (LEAST k. f (Suc \ k) > x)
  { obtain n where x \leq f n
    using assms by (auto simp: filterlim-at-top eventually-at-top-linorder)
    also have f n < f (Suc \ n)
    using assms by (auto simp: strict-mono-Suc-iff)
    finally have \(\exists n. \ f (Suc \ n) > x\) by auto
  }
from LeastI-ex[OF this] have x < f (Suc \ k)
  by (simp add: k-def)
moreover have f k \leq x
proof (cases k)
  case (Suc k')
  have k \leq k' if f (Suc k') > x
  using that unfolding k-def by (rule Least-le)
  with Suc show f k \leq x \(\text{by (cases f k \leq x) (auto simp: not-le)}\)
  qed (use assms in auto)
ultimately show \(?thesis by auto
qed
lemma \(\text{nth-prime-partition}^\prime\):  
assumes \(x \geq 2\)  
shows \(\exists k. x \in \{\text{real (nth-prime } k\}..<\text{real (nth-prime (Suc } k)\}\)  
by (rule strict-mono-sequence-partition)  
(auto simp: strict-mono-Suc-iff assms)  

lemma \(\text{between-nth-primes-imp-nonprime}\):  
assumes \(n > \text{nth-prime } k \land n < \text{nth-prime (Suc } k)\)  
shows \(\neg \text{prime } n\)  
using assms by (metis Suc-leI not-le nth-prime-Suc smallest-prime-beyond-smallest)  

lemma \(\text{nth-prime-partition}''\):  
assumes \(x \geq (2 :: \text{real})\)  
shows \(x \in \{\text{real (nth-prime (nat } \lfloor \pi \ x \rfloor - 1})..<\text{real (nth-prime (nat } \lfloor \pi \ x \rfloor))\}\)  
proof –  
obtain \(n\) where \(n. x \in \{\text{nth-prime } n\}..\text{nth-prime (Suc } n)\)  
using \(\text{nth-prime-partition}^\prime\) assms by auto  
have \(\pi (\text{nth-prime } n) = \pi x\)  
unfolding \(\pi\)-def using between-nth-primes-imp-nonprime \(n\)  
by (intro prime-sum-upto-eqI) (auto simp: le-nat-iff le-floor-iff)  
hence \(\text{real } n = \pi x - 1\)  
by simp  
hence \(n-eq: n = \text{nat } \lfloor \pi x \rfloor - 1\)  
Suc \(n = \text{nat } \lfloor \pi x \rfloor\)  
by linarith+  
with \(\ z\) show \(\text{?thesis}\)  
by simp  
qed

3.4 Relations between different prime-counting functions  

The \(\psi\) function can be expressed as a sum of \(\vartheta\).

lemma \(\psi\)-addef:  
assumes \(x > 0\)  
shows \(\psi x = \text{sum-upto } (\lambda m. \text{prime-sum-upto ln (root m x)) (log } 2 x)\) (is - = ?rhs)  
proof –  
have \(\text{finite}: \text{finite } \{p. \text{prime } p \land \text{real } p \leq y\} \text{ for } y\)  
by (rule finite-subset[of - \{nat \lfloor y \rfloor\}] (auto simp: le-nat-iff le-floor-iff)  
define \(S\) where \(S = (\text{SIGMA } i.\ i. \theta < i \land \text{real } i \leq \text{log } 2 x).\ (p. \text{prime } p \land \text{real } p \leq \text{root } i x)\)  
have \(\psi x = (\sum (p, i). \text{prime } p \land \theta < i \land \text{real } (p ^ i) \leq x.\ ln (\text{real } p))\) unfolding
proof -
  have \(\sum (i, p) : \text{prime} p \land 0 < i \land \text{real } (p^i) \leq x. \ln (p^i)\)
  (auto simp: case_prod-undef mangoldt-non-primepow)
  also have \(\{ (i, p), \text{prime } p \land 0 < i \land \text{real } (p^i) \leq x \} = S\)
  unfolding S-def
proof safe
  fix \(i, p :: \text{nat}\) assume \(i > 0\) real \(i \leq \log 2 x\) prime \(p\) real \(p \leq \sqrt[i]{x}\)
  hence real \((p^i) \leq \sqrt[i]{x \cdot i}\) unfolding of-nat-power by (intro power-mono)
with \(ip\) assms show \(\text{real } (p^i) \leq x\) by simp
next
  fix \(i, p\) assume \(i > 0\) real \(\sqrt[i]{p^i} \leq x\)
  from \(ip\) have \(2^i \leq p^i\) by (intro power-mono) (auto dest: prime-gt-1-nat)
  also have \(\ldots \leq \leq x\) using \(ip\) by simp
finally show \(\text{real } i \leq \log 2 x\)
  using assms by (simp add: le-log-iff powr-realpow)
have \(\sqrt[i]{i} \leq \sqrt[i]{i x}\) using \(ip\) assms
  by (subst real-root-le-iff) auto
also have \(\sqrt[i]{i} \leq \sqrt[i]{i} = \sqrt[i]{p}\)
  using assms \(ip\) by (subst real-root-pos2) auto
finally show \(\text{real } p \leq \sqrt[i]{i \cdot x}\).
qed
also have \(\sum (i,p) \in S. \ln p\) = \text{sum-upto } \(\lambda m. \text{prime-summ-upto } \ln (\text{root } m)\) \(\log 2 x\)
  unfolding \text{sum-upto-def prime-summ-upto-def } S-def using finite by (subst sum.Sigma)
auto
finally show \(\text{thesis}\).
qed

lemma \(\psi\text{-conv-\(\theta\text{-sum} } : \sum x > 0 \implies \psi x = \text{sum-upto } (\lambda m. \theta (\text{root } m)) \) \(\log 2 x\)
by (simp add: \(\psi\text{-altdf } \theta\text{-def}\))

lemma \(\psi\text{-minus-\(\theta} :\)
  assumes \(x : x \geq 2\)
  shows \(\psi x - \theta x = (\sum i \mid 2 \leq i \land \text{real } i \leq \log 2 x. \theta (\text{root } i \cdot x))\)
proof -
  have finite: finite \(\{ i. 2 \leq i \land \text{real } i \leq \log 2 x\}\)
    by (rule finite-subset[of \{2..nat \(\log 2 x\)\}) (auto simp: le-log-iff)
  have \(\psi x = (\sum i \mid 0 < i \land \text{real } i \leq \log 2 x. \theta (\text{root } i \cdot x))\) unfolding x
    by (simp add: \(\psi\text{-conv-\(\theta\text{-sum sum-upto-def}\))
also have \(\{ i. 0 < i \land \text{real } i \leq \log 2 x\} = \text{insert } 1 \{ i. 2 \leq i \land \text{real } i \leq \log 2 x\}\)
  using \(x\)
    by (auto simp: le-log-iff)
also have \((\sum i \mid \theta (\text{root } i \cdot x)) = \theta x =\)
  \((\sum i \mid 2 \leq i \land \text{real } i \leq \log 2 x. \theta (\text{root } i \cdot x))\) using finite
    by (subst sum.insert) auto
finally show \(\text{thesis}\).

63
The following theorems use summation by parts to relate different prime-counting functions to one another with an integral as a remainder term.

**Lemma \( \vartheta \text{-conv-\( \pi \text{-integral} \):}

- **Assumes** \( x \geq 2 \)
- **Shows** \((\lambda t. \pi t / t) \text{ has-integral } (\pi x \ast \ln x - \vartheta x)\) \{2..x\}

**Proof** (cases \( x = 2 \))
- **Case** False
  - **Note** [intro] = \( \text{finite-vimage-real-of-nat-greaterThanAtMost} \)
  - **From** False and **assms** have \( x \): \( x > 2 \) **by** simp
  - **Have** \((\lambda t. \text{sum-upto} (\text{ind prime}) t * (1 / t)) \text{ has-integral}
  - \( \text{sum-upto} (\text{ind prime}) x \ast \ln x - \text{sum-upto} (\text{ind prime}) 2 \ast \ln 2 -
  - \((\sum n \in \text{real} - \{2<..x\}. \text{ind prime} n \ast \ln (\text{real} n)))\) \{2..x\} **using** \( x \)
  - **By** (intro partial-summation-strong**where** \( X = \{\} \))
    - (auto intro: continuous-intros derivative-eq-intros
      simp flip: has-real-derivative-iff-has-vector-derivative)
- **Hence** \((\lambda t. \pi t / t) \text{ has-integral } (\pi x \ast \ln x -
  - (\pi 2 \ast \ln 2 + (\sum n \in \text{real} - \{2<..x\}. \text{ind prime} n \ast \ln n)))\) \{2..x\}
  - **By** (simp add: \( \pi \text{-def prime-sum-upto-altdef1 algebra-simps} \)
    - **Also have** \( \pi 2 \ast \ln 2 + (\sum n \in \text{real} - \{2<..x\}. \text{ind prime} n \ast \ln n) =
    - (\sum n \in \text{insert 2 } (\text{real} - \{2<..x\}). \text{ind prime} n \ast \ln n)\)
      - **By** (subst sum.insert) (auto simp: eval-\( \pi \))
    - **Also have** \( .. = \vartheta x \) unfolding \( \vartheta \text{-def prime-sum-upto-def} \) **using** \( x \)
      - **By** (intro sum.mono-neutral-cong-right) (auto simp: ind-def dest: prime-gt-1-nat)
      - Finally show \( ?\text{thesis} \).

**QED** (auto simp: has-integral-refl eval-\( \pi \) eval-\( \vartheta \))

**Lemma \( \pi \text{-conv-\( \vartheta \text{-integral} \):}

- **Assumes** \( x \geq 2 \)
- **Shows** \((\lambda t. \vartheta t / (t \ast \ln t \sim 2)) \text{ has-integral } (\pi x - \vartheta x / \ln x)\) \{2..x\}

**Proof** (cases \( x = 2 \))
- **Case** False
  - **Define** \( b \) **where** \( b = (\lambda p. \text{ind prime} p \ast \ln (\text{real} p)) \)
  - **Note** [intro] = \( \text{finite-vimage-real-of-nat-greaterThanAtMost} \)
  - **From** False and **assms** have \( x \): \( x > 2 \) **by** simp
    - **Have** \((\lambda t. -(\text{sum-upto} b t \ast (-1 / (t \ast (\ln t)^2)))) \text{ has-integral}
      - -(\text{sum-upto} b x \ast (1 / \ln x) - \text{sum-upto} b 2 \ast (1 / \ln 2) -
      - (\sum n \in \text{real} - \{2<..x\}. b n \ast (1 / \ln (\text{real} n))))\) \{2..x\} **using** \( x \)
      - **By** (intro has-integral-neg partial-summation-strong**where** \( X = \{\} \))
        - (auto intro!: continuous-intros derivative-eq-intros
          simp flip: has-real-derivative-iff-has-vector-derivative simp add: power2-eq-square)
    - **Also have** \( \text{sum-upto} b = \vartheta \)
      - **By** (simp add: \( \vartheta \text{-def b-def prime-sum-upto-altdef1 fun-eq-iff} \)
        - **Also have** \( \vartheta x \ast (1 / \ln x) - \vartheta 2 \ast (1 / \ln 2) -
          - (\sum n \in \text{real} - \{2<..x\}. b n \ast (1 / \ln (\text{real} n))) =
            \vartheta x \ast (1 / \ln x) - (\sum n \in \text{insert 2 } (\text{real} - \{2<..x\}). b n \ast (1 / \ln
            \text{real} n))\)
          - **By** (subst sum.insert) (auto simp: b-def eval-\( \vartheta \))
also have \( \sum_{n \in \text{insert 2} (\text{real} - \{2 < \ldots x\})} b_n \cdot (1 / \ln (\text{real } n)) = \pi x \) using \( x \).

unfolding \( \pi\)-def prime-sum upto-alt def1 sum upto-def

proof (intro sum mono neutral cong left ball I, goal cases)

\( \text{case (3 p)} \)

\( \text{hence } p = 1 \) by auto

\( \text{thus } \{ \text{case by auto} \) qed (auto simp: b-def)

finally show \( ?\text{thesis by simp} \)

qed (auto simp: has integral refl eval \(-\pi\) eval \(-\vartheta\))

lemma integrable weighted \(-\vartheta\):

assumes \( 2 \leq a \leq x \)

shows \( ((\lambda t. \vartheta t / (t * \ln t ^ 2)) \text{ integrable-on } \{a..x\}) \)

proof (cases \( a < x \))

\( \text{case True} \)

hence \( ((\lambda t. \vartheta t * (1 / (t * \ln t ^ 2))) \text{ integrable-on } \{a..x\}) \) using asms

unfolding \( \vartheta\)-def prime sum upto-alt def1

by (intro partial summation integrable strong [where \( X = \{} \text{ and } f = \lambda x. -1 / \ln x \] )

(auto simp flip: has real derivative iff has vector derivative

intro! derivative eq intros continuous intros simp: power2 eq square
field simps)

thus \( ?\text{thesis by simp} \)

qed (insert has integral refl [of - a] asms, auto simp: has integral iff)

lemma \( \vartheta\)-conv \(-\mathbb{M}\)-integral:

assumes \( x \geq 2 \)

shows \( (\mathbb{M} \text{ has integral } (\mathbb{M} x * x - \vartheta x)) \{2..x\} \)

proof (cases \( x = 2 \))

\( \text{case False} \)

with asms have \( x: x > 2 \) by simp

define \( b: \text{nat } \Rightarrow \text{real where } b = (\lambda p. \text{ ind prime } p * \ln p / p) \)

note [intro] = finite vimage real of nat greater than at most

have prime le 2: \( p = 2 \) if \( p \leq 2 \) prime \( p \) for \( p: \text{nat} \)

using that by (auto simp: prime nat iff)

have \( ((\lambda t. \text{ sum upto } b t * 1) \text{ has integral } \text{sum upto } b x * x - \text{sum upto } b 2 * 2 \)

\( \{ \sum n \in \text{real} - \{2 < \ldots x\}, b n * \text{real } n \} \{2..x\} \) using \( x \)

by (intro partial summation strong [of \( \{} \) ]

(auto simp flip: has real derivative iff has vector derivative

intro! derivative eq intros continuous intros)

also have \( \text{sum upto } b = \mathbb{M} \)

by (simp add: fun eq iff primes M def b def prime sum upto alt def1)

also have \( \mathbb{M} * x * x - \mathbb{M} 2 * 2 - (\sum n \in \text{real} - \{2 < \ldots x\}, b n * \text{real } n) = \)

\( \mathbb{M} x * x - (\sum n \in \text{insert 2} (\text{real} - \{2 < \ldots x\}), b n * \text{real } n) \)

by (subst sum insert) (auto simp: eval \(-\mathbb{M}\) b def)

also have \( (\sum n \in \text{insert 2} (\text{real} - \{2 < \ldots x\}), b n * \text{real } n) = \vartheta x \)

65
unfolding $\vartheta$-def prime-sum-upto-def using $x$ by (intro sum.mono-neutral-cong-right) (auto simp: b-def ind-def not-less prime-le-2)

finally show ?thesis by simp

qed (auto simp: eval-$\vartheta$ eval-$\mathfrak{M}$)

lemma $\mathfrak{M}$-conv-$\vartheta$-integral:
assumes $x \geq 2$
shows $(\lambda t. t^2) \text{ has-integral } (\mathfrak{M} x - \vartheta x / x) \{2..\}$
proof (cases $x = 2$)
  case False
  with assms have $x > 2$ by simp
define $b :: \text{nat} \Rightarrow \text{real} \ where \ b = (\lambda p. \text{ind prime } p \ast \ln p)$

note [intro] = finite-vimage-real-of-nat-greaterThanAtMost

have prime-le-2: $p = 2$ if $p \leq 2$ prime $p$ for $p :: \text{nat}$
  using that by (auto simp: prime-nat-iff)

have $(\lambda t. \text{sum-upto } b t * (1 / t^2)) \text{ has-integral }$?
  sum-upto $b x * (-1 / x) - \text{sum-upto } b 2 * (-1 / 2) -$?
  $(\sum \{ n : \text{real } \setminus \{ 2..x \}, b n * (-1 / \text{real } n) \}) \{2..\}$ using $x$
  by (intro partial-summation-strong[of $\{\}$])

(auto simp flip: has-real-derivative-iff-has-vector-derivative simp: power2-eq-square
  intro!: derivative-eq-intros continuous-intros)

also have sum-upto $b = \vartheta$
  by (simp add: fun-eq_iff $\vartheta$-def b-def prime-sum-upto-altdef1)

also have $\vartheta x * (-1 / x) - \vartheta 2 * (-1 / 2) -$?
  $(\sum \{ n : \text{real } \setminus \{ \arctan 2..x \}, b n *$?
  $(-1 / \text{real } n) \}) =$?
  $(-\vartheta x / x - (\sum n : \text{insert 2 } (\text{real } \setminus \{ 2..x \}), b n / \text{real } n))$
  by (subst sum.insert) (auto simp: eval-$\vartheta$ b-def sum-negf)

also have $(\sum n : \text{insert 2 } (\text{real } \setminus \{ 2..x \}), b n / \text{real } n) =$?
  $\mathfrak{M} x$

unfolding primes-$\mathfrak{M}$-def prime-sum-upto-def using $x$
by (intro sum.mono-neutral-cong-right) (auto simp: b-def ind-def not-less prime-le-2)

finally show ?thesis by simp

qed (auto simp: eval-$\vartheta$ eval-$\mathfrak{M}$)

lemma integrable-primes-$\mathfrak{M}$: $\mathfrak{M}$ integrable-on $\{x..y\}$ if $2 \leq x$ for $x y :: \text{real}$

proof
  have $(\lambda x. \mathfrak{M} x * 1) \text{ integrable-on } \{x..y\}$ if $2 \leq x < y$ for $x y :: \text{real}$
  unfolding primes-$\mathfrak{M}$-def prime-sum-upto-altdef1 using that
  by (intro partial-summation-integrable-strong [where $X = \{\}$ and $f = \lambda x. x$])
   (auto simp flip: has-real-derivative-iff-has-vector-derivative
    intro!: derivative-eq-intros continuous-intros)

  thus ?thesis using that-has-integral-refl [of $\mathfrak{M} x$] by (cases $x y$ rule: linorder-cases)

auto

qed

3.5 Bounds

lemma $\vartheta$-upper-bound-coarse:
assumes $x \geq 1$
shows \( \vartheta x \leq x \ast \ln x \)

proof –

have \( \vartheta x \leq \sum\text{upto} (\lambda r. \ln x) x \) unfolding \( \vartheta \)-def prime-sum-upto-altdef1

sum-upto-def

by (intro sum-mono) (auto simp: ind-def)

also have \( \ldots \leq \text{real-of-int} \lfloor x \rfloor \ast \ln x \) using assms

by (simp add: sum-upto-altdef)

also have \( \ldots \leq x \ast \ln x \) using assms by (intro mult-right-mono) auto

finally show \( \?thesis \).

qed

lemma \( \vartheta \leq \psi \): \( \vartheta x \leq \psi x \)

proof (cases \( x \geq 2 \))

case False

hence \( \lfloor x \rfloor = 0 \lor \lfloor x \rfloor = 1 \) by linarith

thus \( \?thesis \) by (auto simp: eval-\vartheta)

next

case True

hence \( \psi x - \vartheta x = (\sum i \mid \lfloor x \rfloor \leq i \land \text{real } i \leq \log 2 x \ast \vartheta (\text{root } i x)) \)

by (rule \( \psi \)-minus-\vartheta)

also have \( \ldots \geq 0 \) by (intro sum-nonneg) auto

finally show \( \?thesis \) by simp

qed

lemma \( \pi \)-upper-bound-coarse:

assumes \( x \geq 0 \)

shows \( \pi x \leq x / 3 + 2 \)

proof –

have \( \{ p. \text{prime } p \land p \leq \lfloor x \rfloor \} \subseteq \{2, 3\} \cup \{ p. \ p \neq 1 \land \text{odd } p \land \neg3 \text{ dvd } p \land p \leq \lfloor x \rfloor \} \)

using primes-dvd-imp-eq[of \( 2 :: \text{nat} \) primes-dvd-imp-eq[of \( 3 :: \text{nat} \) by auto

also have \( \ldots \subseteq \{2, 3\} \cup ((\lambda k. 6 \ast k + 1) \ast \{0 < \ldots \leq \lfloor (x+5)/6 \rfloor \} \cup (\lambda k. 6 \ast k + 5) \ast \{0 < \ldots \leq \lfloor (x+1)/6 \rfloor \}) \)

(is - \cup \?lhs \subseteq - \cup \?rhs)

proof (intro Un-mono subsetI)

fix \( p :: \text{nat} \) assume \( p \in \?lhs \)

hence \( p: p \neq 1 \text{ odd } p \neg3 \text{ dvd } p \ p \leq \lfloor x \rfloor \) by auto

from \( p (1-3) \) have \( \exists k. k > 0 \land p = 6 \ast k + 1 \lor p = 6 \ast k + 5 \) by presburger

then obtain \( k \) where \( k > 0 \land p = 6 \ast k + 1 \lor p = 6 \ast k + 5 \) by blast

hence \( p = 6 \ast k + 1 \land k > 0 \land k < \text{nat } [(x+5)/6] \lor p = 6 \ast k + 5 \land k < \text{nat } [(x+1)/6] \)

unfolding add-divide-distrib using \( p(\?) \) by linarith

thus \( p \in \?rhs \) by auto

qed

finally have subset: \( \{ p. \text{prime } p \land p \leq \lfloor x \rfloor \} \subseteq \ldots \) (is - \subseteq \?A)

have \( \pi x = \text{real } (\card \{ p. \text{prime } p \land p \leq \lfloor x \rfloor \}) \)

by (simp add: \( \pi \)-def prime-sum-upto-altdef2)
also have \( \text{card} \{ p, \text{prime} p \land p \leq \text{nat} \lfloor x \rfloor \} \leq \text{card} \ ?A \)
by \( \text{intro card-mono subset} \) \( \text{auto} \)
also have \( \ldots \leq 2 + (\text{nat} \lfloor (x+5)/6 \rfloor - 1 + \text{nat} \lfloor (x+1)/6 \rfloor) \)
by \( \text{intro order.trans[OF card-Un-le]} \) \( \text{add-mono order.trans[OF card-image-le]} \) \( \text{auto} \)
also have \( \ldots \leq x / 3 + 2 \)
using \( \text{assms unfolding add-divide-distrib by (cases x \geq 1, linarith, simp)} \)
finally show \( \text{thesis by simp} \)
qed

lemma le-numeral-iff: \( m \leq \text{numeral n} \iff m = \text{numeral n} \lor m \leq \text{pred-numeral n} \)
using \( \text{numeral-eq-Suc \ of n by presburger} \)
The following nice proof for the upper bound \( \theta(x) \leq \ln 4 \cdot x \) is taken from Otto Forster’s lecture notes on Analytic Number Theory [4].

lemma prod-primes-upto-less:
\[
\text{defines } F \equiv (\lambda n. (\prod \{ p :: \text{nat}. \text{prime} p \land p \leq n \}))
\]
shows \( n > 0 \implies F n < 4 ^ n \)
proof (induction \( n \) rule: less-induct)
case (less \( n \))
have \( n = 0 \lor n = 1 \lor n = 2 \lor n = 3 \lor \text{even} n \land n \geq 4 \lor \text{odd} n \land n \geq 4 \)
by \( \text{presburger} \)
then consider \( n = 0 \mid n = 1 \mid n = 2 \mid n = 3 \mid \text{even} n \land n \geq 4 \mid \text{odd} n \land n \geq 4 \)
by \( \text{metis} \)
thus \( \text{?case} \)
proof cases
assume \([\text{simp}]: n = 1 \)
have \( *: \{ p, \text{prime} p \land p \leq \text{Suc 0} \} = \{ \} \) by \( \text{auto dest: prime-gt-1-nat} \)
show \( \text{thesis} \) by \( \text{(simp add: F-def \ *)} \)
next
assume \([\text{simp}]: n = 2 \)
have \( *: \{ p, \text{prime} p \land p \leq 2 \} = \{ 2 :: \text{nat} \} \)
by \( \text{auto simp: le-numeral-iff dest: prime-gt-1-nat} \)
thus \( \text{thesis} \) by \( \text{(simp add: F-def \ *)} \)
next
assume \([\text{simp}]: n = 3 \)
have \( *: \{ p, \text{prime} p \land p \leq 3 \} = \{ 2, 3 :: \text{nat} \} \)
by \( \text{auto simp: le-numeral-iff dest: prime-gt-1-nat} \)
thus \( \text{thesis} \) by \( \text{(simp add: F-def \ *)} \)
next
assume \( n: \text{even} n \land n \geq 4 \)
from \( n \) have \( F (n - 1) < 4 ^ (n - 1) \) by \( \text{intro less.IH} \) \( \text{auto} \)
also have \( \text{prime} p \land p \leq n \iff \text{prime} p \land p \leq n - 1 \) for \( p \)
using \( \text{n prime-odd-nat[of n]} \) by \( \text{(cases p = n) auto} \)
hence \( F (n - 1) = F n \) by \( \text{(simp add: F-def)} \)
also have \( 4 ^ (n - 1) \leq (4 ^ n :: \text{nat}) \) by \( \text{(intro power-increasing) auto} \)
finally show \( \text{?case} \).
next
assume \( n \): odd \( n \geq 4 \)
then obtain \( k \) where \( k \)-eq: \( n = \Suc (\odot \cdot k) \) by (auto elim: oddE)
from \( n \) have \( k \): \( k \geq 2 \) unfolding \( k \)-eq by presburger
have prime-dvd: \( p \ dvd \ (n \ choose \ k) \) if \( p: \ prime \ (p \ p) \in \{k+1<..<n\} \) for \( p \)
proof –
  from \( p \ k \ n \) have \( p \ dvd \ pochhammer \ (k + 2) \ k \)
  unfolding pochhammer-prod
    by (subst prime-dvd-prod-iff)
      (auto intro!: bezel[of -p -k -2] simp: k-eq numeral-2-eq-2 Suc-diff-Suc)
  also have pochhammer (real \( (k + 2) \) \( k = \real ((n \ choose \ k) * \text{fact} \ k) \)
    by (simp add: binomial-gbinomial gbinomial-pochhammer k-eq field-simps)
  hence pochhammer \( (k + 2) \ k = (n \ choose \ k) * \text{fact} \ k \)
finally show \( p \ dvd \ (n \ choose \ k) \) using \( p \)
  by (auto simp: prime-dvd-fact-iff prime-dvd-mult-nat)
qed

have \( \prod \{ p: \ prime \ (p \ p) \in \{k+1<..<n\}\} \ dvd \ (n \ choose \ k) \)
proof (rule multiplicity-le-imp-dvd, goal-cases)
  case (\odot \odot \ p)
  thus \( ?\text{case} \)
    case \False
    hence multiplicity \( p \ (\prod \{ p: \ prime \ (p \ p) \in \{k+1<..<n\}\}) = 0 \)
      using \( 2 \)
    by (subst prime-element-multiplicity-prod-distrib) (auto simp: prime-multiplicity-other)
    thus \( ?\text{thesis by auto} \)

next
  case \True
  hence multiplicity \( p \ (\prod \{ p: \ prime \ (p \ p) \in \{k+1<..<n\}\}) = \)
    sum (multiplicity \( p \) \( \{ p: \ prime \ (p \ Suc \ k < p \ p \leq n\} \)
      using \( 2 \)
    by (subst prime-element-multiplicity-prod-distrib) auto
  also have \( \ldots = \sum (\text{multiplicity} \ p) \ \{ p \} \) using \( \True \ 2 \)
  proof (intro sum.zeros_neutral_right ballI)
    fix \( q: \ nat \) assume \( q \in \{ p: \ prime \ (p \ Suc \ k < p \ p \leq n\} - \{ p \}
    hence multiplicity \( p \ q = 0 \)
      using \( 2 \)
    by (cases \( p = q \)) (auto simp: prime-multiplicity-other)
  qed
  also have \( \ldots = 1 \) using \( \True \ 2 \) by simp
  also have \( 1 \leq \text{multiplicity} \ p \ (n \ choose \ k) \)
    using prime-dvd[of \( p \) \( 2 \) \( \True \) by (intro multiplicity-gel) auto
  finally show \( ?\text{thesis} \).
qed

qed auto

hence \( \prod \{ p: \ prime \ (p \ p) \in \{k+1<..<n\}\} \leq (n \ choose \ k) \)
  by (intro dvd-imp-le) (auto simp: k-eq)
also have \( \ldots = 1 / 2 * (\sum \{ i\in\{k, Suc \ k\}, \ n \ choose \ i \}
  using central-binomial-odd[of \( n \)] by (simp add: k-eq)
also have \( (\sum \{ i\in\{k, Suc \ k\}, \ n \ choose \ i \} < (\sum \{ i\in\{0, k, Suc \ k\}, \ n \ choose \ i \)
  using \( k \) by simp

69
also have \( \ldots \leq \left( \sum_{i \leq n} \binom{n}{i} \right) \)
by (intro sum-mono2) (auto simp: k-eq)
also have \( \ldots = (1 + 1) \sim n \)
using binomial[of 1 1 n] by simp
also have \( 1 / 2 \ldots = \text{real} \left( 4 \sim k \right) \)
by (simp add: k-eq power-mult)
finally have less: \( \prod_{\{p \mid \text{prime } p \land p \in \{k + 1 \ldots n\}\}} < 4 \sim k \)
unfolding of-nat-less-iff by simp

have \( \mathcal{F} n = \mathcal{F} (\text{Suc } k) \times \prod_{\{\text{prime } p \land p \in \{k + 1 \ldots n\}\}} \) unfolding F-def
by (subst prod.union-disjoint [symmetric]) (auto intro: prod.cong simp: k-eq)
also have \( \ldots < 4 \sim \text{Suc } k \times 4 \sim k \) using n
by (intro mult-strict-mono less less.IH) (auto simp: k-eq)
also have \( \ldots = 4 \sim (\text{Suc } k + k) \)
by (simp add: power-add)
also have \( \text{Suc } k + k = n \) by (simp add: k-eq)
finally show \( ?\)case.
qed (insert less.prems, auto)

lemma \( \vartheta\)-upper-bound:
assumes \( x \): \( x \geq 1 \)
shows \( \vartheta x < \ln 4 \ast x \)
proof -
  have \( 4 \text{ powr } (\vartheta x / \ln 4) = \prod_{\{\text{prime } p \land p \leq \text{ntr } x\}} 4 \text{ powr } (\log 4 \text{ real } p) \)
  by (simp add: \( \vartheta \)-def powr-sum prime-sum-upto-altdef2 sum-divide-distrib log-def)
  also have \( \ldots = \prod_{\{\text{prime } p \land p \leq \text{ntr } x\}} \text{real } p \)
  by (intro prod.cong) (auto dest: prime-gt-1-nat)
  also have \( \ldots = \text{real} \left( \prod_{\{\text{prime } p \land p \leq \text{ntr } x\}} p \right) \)
  by simp
  also have \( \prod_{\{\text{prime } p \land p \leq \text{ntr } x\}} p < 4 \sim \text{ntr } x \)
  using \( x \) by (intro prod-primes-upto-less) auto
  also have \( \ldots = 4 \text{ powr } \text{real} \left( \text{ntr } x \right) \)
  using \( x \) by (subst powr-realdpow) auto
  also have \( \ldots \leq 4 \text{ powr } x \)
  using \( x \) by (intro powr-mono) auto
  finally have \( 4 \text{ powr } (\vartheta x / \ln 4) < 4 \text{ powr } x \)
  by simp
  thus \( \vartheta x < \ln 4 \ast x \)
  by (subst (asm) powr-less-cancel-iff) (auto simp: field-simps)
qed

lemma \( \vartheta\)-bigo: \( \vartheta \in O(\lambda x. x) \)
by (intro le-imp-bigo-real[of \( \ln 4 \)] eventually-mono[of \( \text{OF eventually-ge-at-top[of 1]\} \)]
less-imp-le[of \( \text{OF \( \vartheta\)-upper-bound\} \)]) auto

lemma \( \psi\)-minus-\( \vartheta\)-bound:
assumes \( x \): \( x \geq 2 \)
shows  \( \psi x - \vartheta x \leq 2 \ln x \cdot \sqrt{x} \)

proof –

have \( \psi x - \vartheta x = (\sum i \mid 2 \leq i \land \text{real } i \leq \log 2 \cdot \vartheta (\sqrt{i} \cdot x)) \) using \( x \)

by (rule \ psi-minus-\vartheta \)

also have \( \ldots \leq (\sum i \mid 2 \leq i \land \text{real } i \leq \log 2 \cdot \ln 4 \cdot \sqrt{i} \cdot x) \) using \( x \)

by (intro \ sum-mono \ less-imp-le \ OF \ \vartheta-upper-bound \) auto

also have \( \ldots \leq (\sum i \mid 2 \leq i \land \text{real } i \leq \log 2 \cdot \ln 4 \cdot \sqrt{i} \cdot x) \) using \( x \)

by (intro \ sum-mono \ mult-mono \) (auto simp: \ le-log-iff \ powr-realpow \ intro!: \ real-root-decreasing \)

also have \( \ldots = \text{card } \{i. \ 2 \leq i \land \text{real } i \leq \log 2 \cdot x\} \) \( \ast \) \( \sqrt{x} \)

by (simp add: \ sqrt-def \)

also have \( \{i. \ 2 \leq i \land \text{real } i \leq \log 2 \cdot x\} = \{2..\text{nat } \lfloor \log 2 \cdot x \rfloor\} \)

by (auto simp: \ le-nat-iff \ le-floor-iff \)

also have \( \log 2 \cdot x \geq 1 \) using \( x \) by (simp add: \ le-log-iff \)

hence \( \text{real } (\text{nat } \lfloor \log 2 \cdot x \rfloor - 1) \leq \log 2 \cdot x \) using \( x \)

by \linarith \)

hence \( \text{card } \{2..\text{nat } \lfloor \log 2 \cdot x \rfloor\} \leq \log 2 \cdot x \) by \simp \)

also have \( \ln (2 \ast 2 :: \text{real}) = 2 \ast \ln 2 \) by (subt \ ln-mul \) auto

hence \( \log 2 \cdot x \ast \ln 4 \ast \sqrt{x} = 2 \ast \ln x \ast \sqrt{x} \) using \( x \)

by (simp add: \ ln-sqrt \ log-def \ powr-2 \ powr-square \ field-simps \)

finally show \( \text{thesis \ using } x \) by (simp add: \ mult-right-mono \)

qed

lemma \( \psi\text{-minus-\vartheta\text{-bigo}: } (\lambda x. \psi x - \vartheta x) \in O(\lambda x. \ln x \ast \sqrt{x}) \)

proof (intro \ bigoI \ [of \ -2 \] \ eventually-mono \ OF \ eventually-ge-at-top \ [of \ 2 \]) \)

fix \( x :: \text{real} \) assume \( x \geq 2 \)

thus \( \text{norm } (\psi x - \vartheta x) \leq 2 \ast \text{norm } (\ln x \ast \sqrt{x}) \)

using \( \psi\text{-minus-\vartheta\text{-bound} \ [of \ x]} \ \vartheta\text{-le-\psi} \ [of \ x] \) by \simp \)

qed

lemma \( \psi\text{-bigo}: \psi \in O(\lambda x. \ x) \)

proof –

have \( (\lambda x. \psi x - \vartheta x) \in O(\lambda x. \ln x \ast \sqrt{x}) \)

by (rule \ psi-minus-\vartheta\text{-bigo} \)

also have \( (\lambda x. \ln x \ast \sqrt{x}) \in O(\lambda x. \ x) \)

by \real-asym \)

finally have \( (\lambda x. \psi x - \vartheta x + \vartheta x) \in O(\lambda x. \ x) \)

by (rule \ sum-in-bigo \) (fact \ \vartheta\text{-bigo} \)

thus \( \text{thesis \ by } \simp \)

qed

We shall now attempt to get some more concrete bounds on the difference between \( \pi(x) \) and \( \theta(x)/\ln x \) These will be essential in showing the Prime Number Theorem later.

We first need some bounds on the integral

\[
\int_{2}^{x} \frac{1}{\ln^2 t} \, dt
\]

in order to bound the contribution of the remainder term. This integral actually has an antiderivative in terms of the logarithmic integral \( \text{li}(x) \), but
since we do not have a formalisation of it in Isabelle, we will instead use the following ad-hoc bound given by Apostol:

**lemma** integral-one-over-log-squared-bound:

assumes $x \geq 4$

shows $\int 2x (\lambda t. 1 / \ln t) \leq \sqrt{x} / (\ln 2 + 4x / \ln x^2)$

**proof** –

from $x$ have $x \ast 1 \leq x \sim 2$ unfolding power2-eq-square by (intro mult-left-mono)

with $x$ have $x': 2 \leq \sqrt{x} x \leq x$

by (auto simp: real-sqrt-le_iff intro: real-le-rsqrt)

have $\int 2x (\lambda t. 1 / \ln t^2) = \int 2(\sqrt{x} x) (\lambda t. 1 / \ln t^2) + \int (\sqrt{x} x) (\lambda t. 1 / \ln t^2)$$

(proof)

(also have $\int I2 \leq \int 2(\sqrt{x} x) (\lambda t. 1 / \ln t^2) \leq \int (\sqrt{x} x) (\lambda t. 1 / \ln t^2)$ using $x'$)

(also have $\int I2 \leq \int (\sqrt{x} x) (\lambda t. 1 / \ln t^2)$ using $x'$)

(also have $\int I2 \leq x \ln 2 \leq 2$ using $x'$)

finally show $\lambda x. x / \ln x^2$ by simp

**qed**

**lemma** integral-one-over-log-squared-bigo:

$(\lambda x. \int 2x (\lambda t. 1 / \ln t^2)) \in O(\lambda x. x / \ln x^2)$

**proof** –

define $ab$ where $ab = (\lambda x. \int 2x (\lambda t. 1 / \ln t^2) + 4x / \ln x^2)$

have eventually $(\lambda x. \int 2x (\lambda t. 1 / (\ln t)^2) \leq |ab x|)$ at-top

using eventually-ge-at-top[of $4$]

(proof)

(also have $\int I2 \leq |ab x|$)

using integral-one-over-log-squared-bound[of $x$] by (simp add: ub-def)

finally show $\lambda x. x / \ln x^2$ unfolding ub-def by real-asym

**qed**

**lemma** $\pi - \theta$-bound:
lemma the following upper bound on $\pi$

As a foreshadowing of the Prime Number Theorem, we can already show

q.e.d.

by \(\text{intro integral-continuous-real continuous-intros}\) auto

have \(r \leq \text{integral } (\lambda t. \ln 4 / \ln t ^ 2)\) unfolding \(r\)-def

using \(\text{integrable-weighted-\(\theta\)-of } 2 \cdot x\) \(\text{integrable[of } \ln 4\]\) \text{assms less-imp-le[OF \(\theta\)-upper-bound]}

by \(\text{intro integral-le divide-right-mono}\) (auto simp: field-simps)

also have \(\ldots = \ln 4 \ast \text{integral } (\lambda t. 1 / \ln t ^ 2)\)

using \(\text{integrable[of } 1\]\) by (subst integral-mult) auto

also have \(\ldots \leq \ln 4 \ast (\sqrt x / \ln 2 ^ 2 + 4 \ast x / \ln x ^ 2)\)

using \text{assms by (intro mult-left-mono integral-one-over-log-squared-bound) auto}

also have \(\ln(\frac{4}{x :: \text{real}}) = 2 \ast \ln 2\)

using \(\ln\text{-realpow[of } 2 \cdot 2\]\) by simp

also have \(\ldots \ast (\sqrt x / \ln 2 ^ 2 + 4 \ast x / \ln x ^ 2) = \text{ub}\)

using \text{assms by (simp add: field-simps power2-eq-square ub-def)}

finally have \(r \leq \ldots\)

moreover have \(r \geq 0\) unfolding \(r\)-def using \text{assms}

by \(\text{intro integral-nonneg integrable-weighted-\(\theta\) divide-nonneg-pos}\) auto

ultimately have \(r \in \{0..\text{ub}\}\) by auto

with \(\pi\text{-conv-\(\theta\)-integral[of } x\]\) \(\text{assms(1)}\) show ?thesis

by \(\text{simp add: } r\text{-def has-integral-iff}\)

q.e.d.

The following statement already indicates that the asymptotics of \(\pi\) and \(\vartheta\) are very closely related, since through it, \(\pi(x) \sim x / \ln x\) and \(\vartheta(x) \sim x\) imply each other.

lemma \(\pi\text{-\(\vartheta\)-bigo}\): \((\lambda x. \pi x - \vartheta x / \ln x) \in O(\lambda x. x / \ln x ^ 2)\)

proof

- define \(\text{ub} = (\lambda x. 2 / \ln 2 \ast \text{sqrt } x + 8 \ast \ln 2 \ast x / \ln x ^ 2)\)

have \((\lambda x. \pi x - \vartheta x / \ln x) \in O(\text{ub})\)

proof (intro le-imp-bigo-real[of 1] eventually-mono[OF eventually-ge-at-top])

fix \(x\) :: real assume \(x \geq 4\)

from \(\pi\text{-\(\vartheta\)-bigo[of this]}\) show \(\pi x - \vartheta x / \ln x \geq 0\) and \(\pi x - \vartheta x / \ln x \leq 1 \ast \text{ub} \cdot x\)

by (simp-all add: ub-def)

qed auto

also have \(\text{ub} \in O(\lambda x. x / \ln x ^ 2)\)

unfolding ub-def by real-asymp

finally show ?thesis.

q.e.d.

As a foreshadowing of the Prime Number Theorem, we can already show the following upper bound on \(\pi(x)\):

lemma \(\pi\text{-upper-bound}:\)

73
assumes \( x \geq (\frac{4}{\ln x}) \)
shows \( \pi x < (\frac{\ln 4 \ln x + 8 \ln 2 x}{\ln x + 2} - 2) / (\ln 2 x) \)

proof –

define \( ub \) where \( ub = \frac{2}{\ln 2} + \sqrt{x} \)

have \( \pi x \leq (\frac{x}{\ln x} + ub) \)
using \( \pi - \vartheta \)-bound [assms unfolding ub-def by simp]
also from assms have \( \vartheta x / (\ln x < (\frac{\ln 4 x}{\ln x} + 2) / (2 \ln x) \)
by (intro \( \vartheta \)-upper-bound divide-strict-right mono) auto
finally show \( \pi x \leq \vartheta x / (\ln x + 2) \)
using assms by (simp add: algebra-simps ub-def)

qed

lemma \( \pi \)-bigo: \( \pi \in O(\frac{x}{\ln x}) \)

proof –

have \( (\lambda x. \pi x - \vartheta x / (\ln x)) \in O(\frac{x}{\ln x}) \)
by (fact \( \pi - \vartheta \)-bigo)
also have \( (\lambda x. \pi x / (\ln x + 2)) \in O(\frac{x}{\ln x}) \)
by real-asym
finally have \( (\lambda x. \pi x - \vartheta x / (\ln x)) \in O(\frac{x}{\ln x}) \)
moreover have eventually \( \lambda x. \pi x - \vartheta x / (\ln x) \) at-top by real-asym
hence eventually \( \lambda x. \pi x / (\ln x) \) at-top by eventually-elim auto
hence \( \lambda x. \vartheta x / (\ln x) \) at-top by real-asym
using \( \vartheta \)-bigo by (intro landau-o-big divide-right)
ultimately have \( (\lambda x. \pi x - \vartheta x / (\ln x + 2 x / (\ln x)) \in O(\frac{x}{\ln x}) \)
by (rule sum-in-bigo)
thus \( \pi \)-bigo by simp

qed

3.6 Equivalence of various forms of the Prime Number Theorem

In this section, we show that the following forms of the Prime Number
Theorem are all equivalent:

1. \( \pi(x) \sim x / \ln x \)
2. \( \pi(x) \ln \pi(x) \sim x \)
3. \( p_n \sim n \ln n \)
4. \( \vartheta(x) \sim x \)
5. \( \psi(x) \sim x \)

We show the following implication chains:

- \((1) \rightarrow (2) \rightarrow (3) \rightarrow (2) \rightarrow (1)\)
- \((1) \rightarrow (4) \rightarrow (1)\)
(4) \rightarrow (5) \rightarrow (4)

All of these proofs are taken from Apostol’s book.

**lemma PNT1-imp-PNT1':**

**assumes** \( \pi \sim [\text{at-top}] (\lambda x. x / \ln x) \)

**shows** \( (\lambda x. \ln (\pi x)) \sim [\text{at-top}] \ln \)

**proof**

1. from `assms` have \( ((\lambda x. \pi x / (x / \ln x)) \rightarrow 1) \) `at-top`
   - by `(rule asymp-equiv-l-strong[OF - eventually-mono[OF eventually-gt-at-top[of I]])` `auto`
2. hence \( ((\lambda x. \ln (\pi x) - \ln x + \ln (\ln x)) \rightarrow 0) \) `at-top`
   - by `(intro filterlim-cong eventually-mono[OF eventually-gt-at-top[of 2]])`
     - `(auto simp: ln field-simps ln-mult π-pos)`
3. finally have \( (\lambda x. \ln (\pi x) - \ln x + \ln (\ln x)) \in o(\lambda x. 1) \)
   - by `(intro smalloI-tendsto)` `auto`
4. also have \( (\lambda::real. \lambda x::real) \in o(\lambda x. \ln x) \)
   - by `(real-asymp)` `auto`
5. finally have \( (\lambda x. \ln (\pi x) - \ln x + \ln (\ln x) - \ln (\ln x)) \in o(\lambda x. \ln x) \)
   - by `(rule sum-in-smallo)` `real-asymp`
6. thus * `(λ x. ln (λ x)) ~ [at-top] ln`
   - by `(simp add: asymp-equiv-altdef)`

**qed**

**lemma PNT1-imp-PNT2:**

**assumes** \( \pi \sim [\text{at-top}] (\lambda x. x / \ln x) \)

**shows** \( (\lambda x. \pi x * \ln (\pi x)) \sim [\text{at-top}] (\lambda x. x) \)

**proof**

1. have \( (\lambda x. \pi x * \ln (\pi x)) \sim [\text{at-top}] (\lambda x. x / \ln x * \ln x) \)
   - by `(intro asymp-equiv-intros assms PNT1-imp-PNT1')`
2. also have \( \ldots \sim [\text{at-top}] (\lambda x. x) \)
   - by `(intro asymp-equiv-refl-ev eventually-mono[OF eventually-gt-at-top[of I]])`
     - `(auto simp: field-simps)`
3. finally have \( (\lambda x. \pi x * \ln (\pi x)) \sim [\text{at-top}] (\lambda x. x) \)
   - by `(simp)`

**qed**

**lemma PNT2-imp-PNT3:**

**assumes** \( (\lambda x. \pi x * \ln (\pi x)) \sim [\text{at-top}] (\lambda x. x) \)

**shows** \( \text{n-thprime} \sim [\text{at-top}] (\lambda n. n * \ln n) \)

**proof**

1. have \( (\lambda n. \text{n-thprime} n) \sim [\text{at-top}] (\lambda n. \pi (\text{n-thprime} n) * \ln (\pi (\text{n-thprime} n))) \)
   - using `assms`
   - by `(rule asymp-equiv-symI[OF asymp-equiv-compose])`
     - `(auto intro!: filterlim-compose[OF filterlim-real-sequentially nth-prime-at-top])`
2. also have \( \ldots = (\lambda n. \text{real} (\text{Suc} n) * \ln (\text{real} (\text{Suc} n))) \)
   - by `(simp add: add-uc)`

75
also have \ldots \sim_{\text{at-top}} (\lambda n. \text{real } n \ast \ln (\text{real } n))
by real-asym
finally show \text{n-thprime} \sim_{\text{at-top}} (\lambda n. n \ast \ln n).
qed

\textbf{lemma \texttt{PNT3-imp-PNT2}:}
\textbf{assumes} \text{n-thprime} \sim_{\text{at-top}} (\lambda n. n \ast \ln n)
\textbf{shows} (\lambda x. \pi x \ast \ln (\pi x)) \sim_{\text{at-top}} (\lambda x. x)
\textbf{proof} (\text{rule asymp-equiv-cover1, rule asymp-equiv-cover-sandwich-real})
show eventually (\lambda x. x \in \{ \text{real } \text{n-thprime} (\text{nat } \lfloor \pi x \rfloor - 1))..\text{real } \text{n-thprime} (\text{nat } \lfloor \pi x \rfloor)\})\}
at-top
using eventually-ge-at-top[of 2]
\textbf{proof} eventually-elim
case (\text{elim } x)
with \text{n-thprime-partition}[\text{of } x] show \text{case by auto}
qed
next
\textbf{have} (\lambda x. \text{real } \text{n-thprime} (\text{nat } \lfloor \pi x \rfloor - 1))) \sim_{\text{at-top}}
(\lambda x. \text{real } (\text{nat } \lfloor \pi x \rfloor - 1) \ast \ln (\text{real } (\text{nat } \lfloor \pi x \rfloor - 1)))
by (\text{rule asymp-equiv-compose}[\text{OF - pi-at-top}, \text{rule asymp-equiv-compose}[\text{OF assms}}]) \text{real-asym}
\textbf{also have} \ldots \sim_{\text{at-top}} (\lambda x. \pi x \ast \ln (\pi x))
by (\text{rule asymp-equiv-compose}[\text{OF - pi-at-top}]) \text{real-asym}
\textbf{finally show} (\lambda x. \text{real } \text{n-thprime} (\text{nat } \lfloor \pi x \rfloor - 1))) \sim_{\text{at-top}} (\lambda x. \pi x \ast \ln (\pi x))
).
next
\textbf{have} (\lambda x. \text{real } \text{n-thprime} (\text{nat } \lfloor \pi x \rfloor))) \sim_{\text{at-top}}
(\lambda x. \text{real } (\text{nat } \lfloor \pi x \rfloor) \ast \ln (\text{real } (\text{nat } \lfloor \pi x \rfloor)))
by (\text{rule asymp-equiv-compose}[\text{OF - pi-at-top}, \text{rule asymp-equiv-compose}[\text{OF assms}}]) \text{real-asym}
\textbf{also have} \ldots \sim_{\text{at-top}} (\lambda x. \pi x \ast \ln (\pi x))
by (\text{rule asymp-equiv-compose}[\text{OF - pi-at-top}]) \text{real-asym}
\textbf{finally show} (\lambda x. \text{real } \text{n-thprime} (\text{nat } \lfloor \pi x \rfloor))) \sim_{\text{at-top}} (\lambda x. \pi x \ast \ln (\pi x))
).
qed

\textbf{lemma \texttt{PNT2-imp-PNT1}:}
\textbf{assumes} (\lambda x. \pi x \ast \ln (\pi x)) \sim_{\text{at-top}} (\lambda x. x)
\textbf{shows} (\lambda x. \ln (\pi x)) \sim_{\text{at-top}} (\lambda x. \ln x)
and \pi \sim_{\text{at-top}} (\lambda x. x / \ln x)
\textbf{proof} –
\textbf{have} ev: eventually (\lambda x. \pi x > 0) \text{-at-top}
eventually (\lambda x. \ln (\pi x) > 0) \text{-at-top}
eventually (\lambda x. \ln (\ln (\pi x)) > 0) \text{-at-top}
by (\text{rule eventually-compose-filterlim}[\text{OF - pi-at-top}, \text{real-asym}])+
let \text{\texttt{diff}} = \lambda x. 1 + \ln (\ln (\pi x)) / \ln (\pi x) - \ln x / \ln (\pi x)
\textbf{have} ((\lambda x. \ln (\pi x) * \text{\texttt{diff}}) \longrightarrow \ln 1) \text{-at-top}
\textbf{proof} (\text{rule Lim-transform-eventually})
from assms have \((\lambda x. \pi x \times \ln (\pi x) / x) \longrightarrow 1\) at-top
by \(\text{rule asymp-equivD-strong}[\text{OF - eventually-mono}[\text{OF eventually-gt-at-top}[\text{OF 1}]]]\) auto

then show \((\lambda x. \ln (\pi x) / \ln (\pi x)) \longrightarrow \ln 1\) at-top
by \(\text{rule tendsto-ln}\) auto

show \(\forall x \in \text{at-top}. \ln (\pi x) / \ln (\pi x) = \ln (\pi x) \ast ?f x\)
using \(\text{eventually-gt-at-top}[\text{of 0}]\) ev
by \(\text{eventually-elim (simp add: field-simps ln-mult ln-div)}\) qed

moreover have \((\lambda x. 1 / \ln (\pi x)) \longrightarrow 0\) at-top
by \(\text{rule filterlim-compose}[\text{OF - \pi-at-top}]\) \text{real-asymp}

ultimately have \((\lambda x. \ln (\pi x) \ast ?f x \ast (1 / \ln (\pi x))) \longrightarrow \ln 1 \ast 0\) at-top
by \(\text{rule tendsto-mult}\)

moreover have eventually \((\lambda x. \ln (\pi x) \ast ?f x \ast (1 / \ln (\pi x)) = ?f x)\) at-top
using \(\text{ev by eventually-elim auto}\)

ultimately have \((?f \longrightarrow \ln 1 \ast 0)\) at-top
by \(\text{rule Lim-transform-eventually}\)

hence \((\lambda x. 1 + \ln (\ln (\pi x)) / \ln (\pi x) - ?f x) \longrightarrow 1 + 0 - \ln 1 \ast 0\) at-top
by \(\text{intro tendsto-intros filterlim-compose}[\text{OF - \pi-at-top}]\) \(\text{real-asymp | simp}\)+

hence \((\lambda x. \ln x / \ln (\pi x)) \longrightarrow 1\) at-top
by \(\text{simp}\)

thus \(*: (\lambda x. \ln (\pi x)) \sim[\text{at-top}] (\lambda x. \ln x)\)
by \(\text{rule asymp-equiv-symI}[\text{OF asymp-equivI}]\)

have eventually \((\lambda x. \pi x = \pi x \ast \ln (\pi x) / \ln (\pi x))\) at-top
using \(\text{ev by eventually-elim auto}\)

hence \(\pi \sim[\text{at-top}] (\lambda x. \pi x \ast \ln (\pi x) / \ln (\pi x))\)
by \(\text{rule asymp-equiv-refl-ev}\)

also from assms and \(* have \((\lambda x. \pi x \ast \ln (\pi x) / \ln (\pi x)) \sim[\text{at-top}] (\lambda x. x / \ln x)\)
by \(\text{rule asymp-equiv-intros}\)

finally show \(\pi \sim[\text{at-top}] (\lambda x. x / \ln x)\).

qed

lemma \text{PNT4-imp-PNT5}:
assumes \(\emptyset \sim[\text{at-top}] (\lambda x. x)\)
shows \(\psi \sim[\text{at-top}] (\lambda x. x)\)

proof –

define \(r\) where \(r = (\lambda x. \psi x - \emptyset x)\)

have \(r \in \text{O}(\lambda x. \ln x \ast \sqrt{x})\)

unfolding \(\text{r-def}\) by \(\text{fact \psi-minus-\emptyset-bigo}\)

also have \((\lambda x::\text{real.} \ln x \ast \sqrt{x}) \in \text{o}(\lambda x. x)\)

by \text{real-asymp}

finally have \(r: r \in \text{o}(\lambda x. x)\).

have \((\lambda x. \emptyset x + r \ast x) \sim[\text{at-top}] (\lambda x. x)\)

using \(\text{assms \ r by (subst asymp-equiv-add-right) auto}\)

thus \(?\text{thesis by (simp add: r-def)}\)

qed
lemma PNT4-imp-PNT1:
assumes \( \vartheta \sim_{\text{at-top}} (\lambda x. \ x) \)
shows \( \pi \sim_{\text{at-top}} (\lambda x. \ x / \ln x) \)
proof
  have \( (\lambda x. \ (x - \vartheta \ x / \ln x)) + ((\vartheta \ x - \ x) / \ln x)) \in o(\lambda x. \ x / \ln x) \)
  proof (rule sum-in-smallo)
    have \( (\lambda x. \ (x - \vartheta \ x / \ln x)) \in O(\lambda x. \ x / \ln x - 2) \)
    by (rule \( \pi - \vartheta \times \text{-bigo} \))
    also have \( (\lambda x. \ x / \ln x - 2) \in o(\lambda x. \ x / \ln x :: \text{real}) \)
    by real-asym
  finally show \( (\lambda x. \ x - \vartheta \ x / \ln x) \in o(\lambda x. \ x / \ln x) \).
next
  have eventually \( (\lambda x::\text{real}, \ x > 0) \text{-at-top} \) by real-asym
  hence eventually \( (\lambda x::\text{real}, \ x \neq 0) \text{-at-top} \) by eventually-elim auto
  thus \( (\lambda x. \ (\vartheta \ x - \ x) / \ln x) \in o(\lambda x. \ x / \ln x) \)
  by (intro landau-o.small.divide-right asymp-equiv-imp-diff-smallo assms)
  qed
  thus \( ? \text{thesis} \) by (simp add: diff-distribute asymp-equiv-asymp-altdef)
  qed

lemma PNT1-imp-PNT4:
assumes \( \lambda x. \ x / \ln x \)
shows \( \vartheta \sim_{\text{at-top}} (\lambda x. \ x) \)
proof
  have \( (\lambda x. \ x - \vartheta \ x / \ln x) \in \Theta(\lambda x. \ - ((\pi \ x - \vartheta \ x / \ln x) \times \ln x)) \)
  by (intro bitheta-cong eventually-mono[OF eventually-gt-at-top[of 1]])
  (auto simp: field-simps)
  also have \( (\lambda x. \ - ((\pi \ x - \vartheta \ x / \ln x) \times \ln x)) \in O(\lambda x. \ x / (\ln x)^2 \times \ln x) \)
  unfolding landau-o.big.uminus-in-jiff by (intro landau-o.big.mult-right \( \pi - \vartheta \times \text{-bigo} \))
  also have \( (\lambda x::\text{real}, \ x / (\ln x)^2 \times \ln x) \in o(\lambda x. \ x / \ln x) \times \ln x) \)
  by real-asym
  also have \( (\lambda x. \ x / \ln x \times \ln x) \in \Theta(\lambda x. \ x \times \ln x) \)
  by (intro asymp-equiv-imp-bitheta asymp-equiv-intros asymp-equiv-symI[OF assms])
  finally show \( (\lambda x. \ x - \pi \ x / \ln x) \in o(\lambda x. \ x \times \ln x) \).
  qed
  also have \( \ldots \sim_{\text{at-top}} (\lambda x. \ x \times \ln x) \)
  by (intro asymp-equiv-intros assms)
  also have \( \ldots \sim_{\text{at-top}} (\lambda x. \ x) \)
  by real-asym
  finally show \( ? \text{thesis} \).
  qed

lemma PNT5-imp-PNT4:
assumes \( \psi \sim_{\text{at-top}} (\lambda x. \ x) \)
shows \( \vartheta \sim_{\text{at-top}} (\lambda x. \ x) \)
proof
  
define \( r \) where \( r = (\lambda x. \vartheta x - \psi x) \)
  
have \((\lambda x. \psi x - \vartheta x) \in O(\lambda x. \ln x * \sqrt{x})\)
    
by \( (\text{fact \( \psi \)-minu-\( \vartheta \)-bigo}) \)
  
also have \((\lambda x. \psi x - \vartheta x) = (\lambda x. -r x)\)
    
by \( (\text{simp add: \( r \)-def}) \)
  
finally have \( r \in O(\lambda x. \ln x * \sqrt{x})\)
    
by \( \text{simp} \)
  
also have \((\lambda x::real. \ln x * \sqrt{x}) \in o(\lambda x. x)\)
    
by \( \text{real-asymp} \)
  
finally have \( r: r \in o(\lambda x. x) \).

have \((\lambda x. \psi x + r x) \sim \[\text{at-top}\] (\lambda x. x)\)
  
using \( \text{assms \( r \)} \) by \( (\text{subst \( \text{asymp-equiv-add-right}\}) \) auto \)
  
thus \( ?\text{thesis} \) by \( (\text{simp add: \( r \)-def}) \)

qed

3.7 The asymptotic form of Mertens’ First Theorem

Mertens’ first theorem states that \( \mathcal{M}(x) - \ln x \) is bounded, i.e. \( \mathcal{M}(x) = \ln x + O(1) \).

With some work, one can also show some absolute bounds for \( |\mathcal{M}(x) - \ln x| \), and we will, in fact, do this later. However, this asymptotic form is somewhat easier to obtain and it is (as we shall see) enough to prove the Prime Number Theorem, so we prove the weak form here first for the sake of a smoother presentation.

First of all, we need a very weak version of Stirling’s formula for the logarithm of the factorial, namely:

\[
\ln(|x|!) = \sum_{n \leq x} \ln x = x \ln x + O(x)
\]

We show this using summation by parts.

lemma \( \text{stirling-weak:} \)

assumes \( x: x \geq 1 \)

shows \( \text{sum-upto} \ \ln x \in \{x * \ln x - x - \ln x + 1 .. x * \ln x\} \)

proof (cases \( x = 1 \))
  
  case True
  
  have \( \{0<..<\text{Suc 0}\} = \{1\} \) by auto
  
  with True show \( ?\text{thesis} \) by \( (\text{simp add: \text{sum-upto-altdef})} \)
  
next
  
  case False
  
  with \( \text{assms} \) have \( x: x > 1 \) by simp
  
  have \((\lambda t. \text{sum-upto} (\lambda x. t * (1 / t)) \text{ has-integral}\)
    
    \( \text{sum-upto} (\lambda x. t) x * \ln x - \text{sum-upto} (\lambda x. t) 1 * \ln 1 - (\sum_{n \in \text{real}} \text{'} \{1<..<x\}. 1 * \ln (\text{real } n))\) \( \{1..x\} \text{ using} \)
    
    \( \text{by} (\text{intro \text{partial-summation-strong} [of \}])\)
      
      \( \text{(auto simp \text{flip: \text{has-real-derivative-if-f-has-vector-derivative})} \))

79
lemma stirling-weak-bigo: \((\lambda x::\text{real}. \sum\text{upto } \ln x \leq \ln x + \ln x) \in O(\lambda x. x)\)
proof
  have \((\lambda x. \sum\text{upto } \ln x - x * \ln x) \in O(\lambda x. -(\sum\text{upto } \ln x - x * \ln x))\)
  by (subt landau-o.biguminus) auto
  also have \((\lambda x. -(\sum\text{upto } \ln x - x * \ln x)) \in O(\lambda x. x + \ln x - 1)\)
  proof (intro le-imp-biga-real[of 2] eventually- mono[OF eventually-ge-at-top[of 1]], goal-cases)
    case (2 x)
    thus \(\text{case using stirling-weak[of x] by (auto simp: algebra-simps)}\) 
  next
    case (3 x)
    thus \(\text{case using stirling-weak[of x] by (auto simp: algebra-simps)}\) 
  qed auto
  also have \((\lambda x. x + \ln x - 1) \in O(\lambda x::\text{real}. x)\) by real-asymp
  finally show \(\text{thesis} \) .
qed
The key to showing Mertens’ first theorem is the function

\[ h(x) := \sum_{n \leq x} \frac{\Lambda(d)}{d} \]

where \( \Lambda \) is the Mangoldt function, which is equal to \( \ln p \) for any prime power \( p^k \) and 0 otherwise. As we shall see, \( h(x) \) is a good approximation for \( \Re(x) \), as the difference between them is bounded by a constant.

**Lemma** sum-upto-mangoldt-over-id-minus-phi-bounded:

\((\lambda x. \text{sum-upto}(\lambda d. \text{mangoldt}(d) / \text{real}(d)) \cdot (x - \Re(x))) \in O(\lambda x. 1)\)

**Proof**

- **Define** \( f \) where \( f = (\lambda d. \text{mangoldt}(d) / \text{real}(d)) \)
- **Define** \( C \) where \( C = (\sum p. \ln(\text{real}(p + 1)) \cdot (1 / \text{real}(p * (p - 1)))) \)
- **Have** summable:summable \((\lambda p:nat. \ln(p + 1) \cdot (1 / (p * (p - 1))))\)
- **Proof** (rule summable-comparison-test-big)
  - **Show** summable \((\lambda p:nat. \ln(\text{real}(p + 1)) \cdot (1 / (p * (p - 1))))\)
  - **By** (simp add: summable-real-powr-iff)

**Qed**

**Have** diff-bound: \( \text{sum-upto} f x \cdot \Re x \in \{0..C\} \) if \( x \geq 4 \) for \( x \)

**Proof**

- **Define** \( S \) where \( S = \{(p, i). \text{prime} p \land 0 < i \land \text{real}(p \sim i) \leq x\} \)
- **Define** \( S' \) where \( S' = (\Sigma p. \text{nat} \cdot \text{log} 2 x) \times \{(x, 1)\} \)
- **Have** \( S \subseteq \{0 < i \} \times \{0 < i \} \) using x primepows-le-subset[of x 1] by (auto simp: Suc-le-eq)
- **Hence** finite \( S \) by (rule finite-subset) auto
- **Note** \( \text{fin = finite-subset[OF - this, unfolded S-def]} \)

**Have** \( \text{sum-upto} f x = \sum_{(p, i) \in S} \ln(\text{real}(p)) / \text{real}(p \sim i) \) unfolding S-def
  - (intro sum-upto-primepows) (auto simp: f-def mangoldt-non-primepow)
  - Also have \( S = \{(x, 1): \text{prime} p \land p \leq x \} \times \{1\} \cup \{(x, i): \text{prime} p \land 1 < i \land \text{real}(p \sim i) \leq x\} \)
  - By (auto simp: S-def not-less le-Suc-eq not-le intro: Suc-lessI)
  - Also have \( \sum_{(p, i) \in \ldots} \ln(\text{real}(p)) / \text{real}(p \sim i) = \sum_{(p, i) \in \{p. \text{prime} p \land \text{af-nat} p \leq x\} \times \{1\}} \ln(\text{real}(p)) / \text{real}(p \sim i) \)
  - By (auto simp: S-def not-less le-Suc-eq not-le intro: Suc-lessI)
  - Also have \( \sum_{(p, i) \sim \ldots} \ln(\text{real}(p)) / \text{real}(p \sim i) = \sum_{(p, i) \sim \ldots} \ln(\text{real}(p)) / \text{real}(p \sim i) \)
    - (is = ?S1 + ?S2)
Next, we show that our \( h(x) \) itself is close to \( \ln x \), i.e.:

\[
\sum_{n \leq x} \frac{A(d)}{d} = \ln x + O(1)
\]

** lemma sum-upto-mangoldt-over-id-asymptotics:
(\lambda x. \text{sum-upto} (\lambda d. \text{mangoldt} d / \text{real} d) x - \ln x) \in O(\lambda x. 1)

proof
- define \( r \) where \( r = (\lambda n::\text{real}. \text{sum-upto} (\lambda d. \text{mangoldt} d * (n / d - \text{real-of-int} [n / d])) n) \)
  have \( r : r \in O(\psi) \)
  proof (intro landau-o.bigl[of 1] eventually-mono[OF eventually-ge-at-top[of 0]])
    fix \( x :: \text{real} \) assume \( x : x \geq 0 \)
    have eq: \( \{1..\text{nat} \lfloor x \rfloor\} = \{0<..\text{nat} \lfloor x \rfloor\} \) by auto
    hence \( r x \geq 0 \) unfolding r-def sum-upto-def
      (auto simp: floor-le-iff)
    moreover have \( x / \text{real} d \leq 1 + \text{real-of-int} [x / \text{real} d] \) for \( d \) by linarith
    hence \( r x \leq \text{sum-upto} (\lambda d. \text{mangoldt} d * 1) x \) unfolding sum-upto-altdef eq
    r-def using x
      by (intro sum-mono mult-mono mangoldt-nonneg)
      (auto simp: less_imp_le[OF frac_lt_1] algebra_simps)
    ultimately show \( \text{norm} (r x) \leq 1 * \text{norm} (\psi x) \) by (simp add: psi-def)
  qed auto
  also have \( \psi \in O(\lambda x. x) \) by (fact psi-bigo)
  finally have \( r : r \in O(\lambda x. x) \).

define \( r' \) where \( r' = (\lambda x::\text{real}. \text{sum-upto} \ln x - x * \ln x) \)
have \( r'-\text{bigo} : r' \in O(\lambda x. x) \)
  using stirling-weak-bigo unfolding r'-def .
have ln-fact: \( \ln (\text{fact} n) = (\sum d=1..n. \ln d) \) for \( n \)
  by (induction \( n \)) (simp-all add: ln-mult)
have \( r' : \text{sum-upto} \ln n = n * \ln n + r' n \) for \( n :: \text{real} \)
  unfolding r'-def sum-upto-altdef by (auto intro!: sum.cong)
have eventually (\lambda n. \text{sum-upto} (\lambda d. \text{mangoldt} d / d) n - \ln n = r' n / n + r n / n) at-top
  using eventually_gt_at_top
proof eventually-elim
  fix \( x :: \text{real} \) assume \( x : x > 0 \)
  have sum-upto-ln x = sum-upto (\lambda n. \text{mangoldt} n * (\text{real} (\text{nat} \lfloor x / n \rfloor))) x
    unfolding sum-upto-ln_cone-sum-upto-mangoldt ..
  also have \( \ldots = \text{sum-upto} (\lambda d. \text{mangoldt} d * (x / d)) x - r x \)
  unfolding sum-upto-def by (simp add: algebra_simps sum_subtatcf r-def sum-upto-def)
  also have \( \text{sum-upto} (\lambda d. \text{mangoldt} d * (x / d)) x = x * \text{sum-upto} (\lambda d. \text{mangoldt} d / d) x \)
    unfolding sum-upto-def by (subst sum-distrib_left) (simp add: field_simps)
  finally have \( x * \text{sum-upto} (\lambda d. \text{mangoldt} d / \text{real} d) x = r' x + r x + x * \ln x \)
    by (simp add: r' algebra_simps)
  thus \( \text{sum-upto} (\lambda d. \text{mangoldt} d / d) x - \ln x = r' x / x + r x / x \)
    using x by (simp add: field_simps)
qed
hence \( (\lambda x. \text{sum-upto} (\lambda d. \text{mangoldt} d / d) x - \ln x) \in \Theta(\lambda x. r' x / x + r x / x) \)
by (rule bigthetaI-cong)
also have $(\lambda x. r' x / x + r x / x) \in O(\lambda -. 1)$
by (intro sum-in-bigo) (insert r r'-bigo, auto simp: landau-divide-simps)
finally show ?thesis .

qed

Combining these two gives us Mertens’ first theorem.

**Theorem** mertens-bounded: $(\lambda x. \mathcal{M} x - \ln x) \in O(\lambda -. 1)$
**Proof** –
define $f$ where $f = \sum_{d} (\lambda d. \text{mangoldt} d / d)$
have $(\lambda x. (f x - \ln x) - (f x - \mathcal{M} x)) \in O(\lambda -. 1)$
using sum-upto-mangoldt-over-id-asymptotics
sum-upto-mangoldt-over-id-minus-phi-bounded
unfolding f-def by (rule sum-in-bigo)
thus ?thesis by simp

qed

**Lemma** primes-M-bigo: $\mathcal{M} \in O(\lambda x. \ln x)$
**Proof** –
have $(\lambda x. \mathcal{M} x - \ln x) \in O(\lambda -. 1)$
by (rule mertens-bounded)
also have $(\lambda :: \text{real}. 1) \in O(\lambda x. \ln x)$
by real-asymptotic
finally have $(\lambda x. \mathcal{M} x - \ln x + \ln x) \in O(\lambda x. \ln x)$
by (rule sum-in-bigo) auto
thus ?thesis by simp

qed

end

4 The Prime Number Theorem

**Theory** Prime-Number-Theorem
**Imports**
Newman-Ingham-Tauberian
Prime-Counting-Functions

begin

4.1 Constructing Newman’s function

Starting from Mertens’ first theorem, i.e. $\mathcal{M}(x) = \ln x + O(1)$, we now want to derive that $\mathcal{M}(x) = \ln x + c + o(1)$. This result is considerably stronger and it implies the Prime Number Theorem quite directly.

In order to do this, we define the Dirichlet series

$$f(s) = \sum_{n=1}^{\infty} \frac{\mathcal{M}(n)}{n^s}.$$
We will prove that this series extends meromorphically to \( \Re(s) \geq 1 \) and apply Ingham’s theorem to it (after we subtracted its pole at \( s = 1 \)).

**Definition** *fds-newman* where

\[
\text{fds-newman} = \text{fds} (\lambda n. \text{complex-of-real} (\Re n))
\]

**Lemma** *fds-nth-newman*:

\[
fds-nth \text{fds-newman} n = \text{of-real} (\Re n)
\]

**Lemma** *norm-fds-nth-newman*:

\[
\text{norm} (fds-nth fds-newman n) = \Re n
\]

**Proof**

\[
\begin{align*}
\text{The Dirichlet series } f(s) + \zeta'(s) \text{ has the coefficients } \Re(n) - \ln n, \text{ so by Mertens’ first theorem, } f(s) + \zeta'(s) \text{ has bounded coefficients.}
\end{align*}
\]

**Lemma** *bounded-coeffs-newman-minus-deriv-zeta*:

**Defines** \( f \equiv \text{fds-newman} + \text{fds-deriv fds-zeta} \)

**Shows** \( \text{Bseq} (\lambda n. \text{fds-nth f n}) \)

**Proof**

\[
\begin{align*}
\text{have } (\lambda n. \Re (\text{real n}) - \ln (\text{real n})) &\in O(\lambda-. 1) \\
\text{using mertens-bounded by (rule landau-o.big.compose) real-asymp}
\end{align*}
\]

\[
\begin{align*}
\text{from natfun-bigo-1E[OF this, of 1]} \\
\text{obtain } c \text{ where } c \geq 1 \land n. \Re (\text{real n}) - \ln (\text{real n}) \leq c \text{ by auto}
\end{align*}
\]

**Show** \( ?\text{thesis} \)

**Proof**

\[
\begin{align*}
\text{fix } n :: \text{nat} \\
\text{show } \text{norm} (\text{fds-nth f n}) \leq c
\end{align*}
\]

\[
\begin{align*}
\text{proof (intro BseqI[of c] allI)} \\
\text{fix } n :: \text{nat} \\
\text{show } \text{norm} (\text{fds-nth f n}) \leq c
\end{align*}
\]

\[
\begin{align*}
\text{case False} \\
\text{hence } \text{fds-nth f n} = \text{of-real} (\Re n - \ln n)
\end{align*}
\]

\[
\begin{align*}
\text{by } (\text{simp add; f-def fds-nth-newman \text{fds-nth-deriv fds-zeta scaleR-conv-of-real})}
\end{align*}
\]

\[
\begin{align*}
\text{also from } (n \neq 0) \text{ have norm } \ldots \leq c \\
\text{using } c(2)[\text{of n}] \text{ by (simp add; in-Reals-norm)}
\end{align*}
\]

\[
\begin{align*}
\text{finally show } ?\text{thesis}. \\
\text{qed (insert c, auto)}
\end{align*}
\]

\[
\begin{align*}
\text{qed (insert c, auto)}
\end{align*}
\]

\[
\text{qed}
\]

A Dirichlet series with bounded coefficients converges for all \( s \) with \( \Re(s) > 1 \) and so does \( \zeta'(s) \), so we can conclude that \( f(s) \) does as well.

**Lemma** *abs-conv-abscissa-newman*:

\[
\text{abs-conv-abscissa fds-newman} \leq 1
\]

**And**

\[
\text{conv-abscissa-newman} \leq 1
\]

**Proof**

\[
\begin{align*}
\text{define } f \text{ where } f = \text{fds-newman} + \text{fds-deriv fds-zeta} \\
\text{have abs-conv-abscissa } f \leq 1 \\
\text{using bounded-coeffs-newman-minus-deriv-zeta unfolding f-def}
\end{align*}
\]
by (rule bounded-coffs-imp-abs-conv-ambscissa-le-1)
hence \( \text{abs-conv-ambscissa} (f - \text{fds-deriv} \text{fds-zeta}) \leq 1 \)
by (intro abs-conv-ambscissa-diff-le1) (auto simp: abs-conv-ambscissa-deriv)
also have \( f - \text{fds-deriv} \text{fds-zeta} = \text{fds-newman} \) by (simp add: f-def)
finally show \( \text{abs-conv-ambscissa} \text{fds-newman} \leq 1 \).
from conv-le-abs-conv-ambscissa and this show \( \text{conv-ambscissa} \text{fds-newman} \leq 1 \)
by (rule order.trans)
qed

We now change the order of summation to obtain an alternative form of \( f(s) \) in terms of a sum of Hurwitz \( \zeta \) functions.

**lemma eval-fds-newman-conv-infsetsum:**

assumes \( \text{Re } s > 1 \)

shows \( \text{eval-fds} \text{fds-newman} s = (\sum_{a \mid \text{prime } p} (\ln (\text{real } p) / \text{real } p) * \text{hurwitz-zeta} p s) \)

\( (\lambda p. \ln (\text{real } p) / \text{real } p * \text{hurwitz-zeta} p s) \) \text{abs-summable-on} \ \{ p. \text{prime } p \}

**proof**

from \( s \) have \( \text{conv: fds-conv-converges} \text{fds-newman} s \)
by (intro fds-conv-converges le-less-trans[OF abs-conv-ambscissa-newman]) auto

define \( f \) where \( f = (\lambda n. \ln (\text{real } p) / \text{real } p / \text{of-nat } n \text{ powr } s) \)

have eq: \( (\sum_{a \in\{p..\}}. f n p) = \ln (\text{real } p) / \text{real } p * \text{hurwitz-zeta} p s \) if prime p
for p

**proof**

have \( (\sum_{a \in\{p..\}}. f n p) = (\sum_{a \in\{p..\}}. (\ln (\text{real } p) / \text{of-nat } p) * (1 / \text{of-nat } x \text{ powr } s)) \)

by (simp add: f-def)
also have \( \ldots = (\ln (\text{real } p) / \text{of-nat } p) * (\sum_{a \in\{p..\}}. 1 / \text{of-nat } x \text{ powr } s) \)
using \( \text{abs-summable-hurwitz-zeta}[of s 0 p] \) that s
by (intro infsetsum-cmult-right) (auto dest: prime-gt-zero)
also have \( (\sum_{a \in\{p..\}}. 1 / \text{of-nat } x \text{ powr } s) = \text{hurwitz-zeta} p s \)
using \( s \) that by (subst hurwitz-zeta-conv-infsetsum(2))
(auto dest: prime-gt-zero simp: field-simps powr-minus)

finally show \( \text{thesis} \).

qed

have norm-f: \( \text{norm} (f n p) = \ln p / p / n \text{ powr } \text{Re } s \) if prime p for n p :: nat
by (auto simp: fds-def-norm-conv-norm-mult norm-powr-real-powr)
from conv have \( (\lambda n. \text{norm} (\text{fds-nth} \text{fds-newman } n / \text{of-nat } s)) \) \text{abs-summable-on} \text{UNIV}

by (intro abs-summable-on-normI) (simp add: fds-abs-converges-altdef)
also have \( (\lambda n. \text{norm} (\text{fds-nth} \text{fds-newman } n / \text{of-nat } s)) =
(\lambda n. \sum p \mid \text{prime } p \land p \leq n. \text{norm} (f n p)) \)
by (auto simp: norm-divide norm-fds-nth-newman sum-divide-distrib primes-M-def prime-sum-upto-def norm-mult norm-f powr-powr intro!: sum.cong)

finally have summable1: \( (\lambda(n,p). f n p) \) \text{abs-summable-on} \( (\Sigma\Sigma n:\text{UNIV}. \{ p. \text{prime } p \land p \leq n \}) \)
using conv by (subst abs-summable-on-Sigma-iff) auto
also have \(\text{this} \iff (\lambda(p,n), f n p) \text{ abs-summable-on}�\)
\((\lambda(n,p), (p,n)) \cdot (\Sigma n:UNIV. \{p.\ \text{prime} p \land p \leq n\})\)

by (\(\text{subst abs-summable-on-reindex-iff \{symmetric\}}\) (auto simp: case-prod-unfold inj-on-def)

also have \((\lambda(n,p), (p,n)) \cdot (\Sigma n:UNIV. \{p.\ \text{prime} p \land p \leq n\}) = \Sigma p:\{p.\ \text{prime} p\}. \{p..\}\) by auto

finally have \(\text{summable2:} (\lambda(p,n), f n p) \text{ abs-summable-on} \ldots . \)
from \(\text{abs-summable-on-Sigma-project1}^{[\text{OF this}]}\)
have \((\lambda p. \Sigma a n \in\{p..\}, f n p) \text{ abs-summable-on} \{p.\ \text{prime} p\}\) by auto
also have \(\text{this} \iff (\lambda p. \ln (\text{real} p) / \text{real} p * \text{hurwitz-zeta} p s) \text{ abs-summable-on}�\)
\(\{p.\ \text{prime} p\}\)

by (intro abs-summable-on-cong eq) auto

finally show \(\ldots . \)

have \(\text{eval-fds fds-neuman} s = \)
\((\Sigma a n, \Sigma p | \text{prime} p \land p \leq n. \ln (\text{real} p) / \text{real} p / \text{of-nat} n \text{ powr} s)\)

using \(\text{conv} \) by (simp add: \(\text{eval-fds-altdefs fds-nth-neuman sum-divide-distrib primes-M-def prime-sum-upto-def})

also have \(\ldots . = (\Sigma a n, \Sigma a p | \text{prime} p \land p \leq n. \ f n p)\)

unfolding \(\text{f-def} \) by (\(\text{subst infsetsum-finite})\) auto
also have \(\ldots . = (\Sigma a (n,p), p) \in (\Sigma n:UNIV. \{p.\ \text{prime} p \land p \leq n\}). \ f n p\)

using \(\text{summable1 by (subst infsetsum-Sigma)) auto}\)
also have \(\ldots . = (\Sigma a (n,p), n) \in (\lambda(n,p), (p,n)) \cdot (\Sigma n:UNIV. \{p.\ \text{prime} p \land p \leq n\}). \ f n p\)

by (\(\text{subst infsetsum-reindex})\) (auto simp: case-prod-unfold inj-on-def)
also have \((\lambda(n,p), (p,n)) \cdot (\Sigma n:UNIV. \{p.\ \text{prime} p \land p \leq n\}) = \Sigma p:\{p.\ \text{prime} p\}. \{p..\}\) by auto
also have \((\Sigma a(n,p) \in\ldots f n p) = (\Sigma a p | \text{prime} p. \Sigma a n \in\{p..\}). \ f n p)\)

using \(\text{summable2 by (subst infsetsum-Sigma)) auto}\)
also have \((\Sigma a p | \text{prime} p. \Sigma a n \in\{p..\}). \ f n p) = \)
\((\Sigma a p | \text{prime} p. \ln (\text{real} p) / \text{real} p * \text{hurwitz-zeta} p s)\)

by (\(\text{intro infsetsum-cong eq})\) auto
finally show \(\text{eval-fds fds-neuman} s = \)
\((\Sigma a p | \text{prime} p. \ln (\text{real} p) / \text{real} p * \text{hurwitz-zeta} p s)\).

qed

We now define a meromorphic continuation of \(f(s)\) on \(\Re(s) > \frac{1}{2}\).
To construct \(f(s)\), we express it as

\[
    f(s) = \frac{1}{z - 1} \left( \tilde{f}(s) - \frac{\zeta'(s)}{\zeta(s)} \right),
\]

where \(\tilde{f}(s)\) (which we shall call \text{pre-neuman}) is a function that is analytic on \(\Re(s) > \frac{1}{2}\), which can be shown fairly easily using the Weierstraß M test. \(\zeta'(s)/\zeta(s)\) is meromorphic except for a single pole at \(s = 1\) and one \(k\)-th order pole for any \(k\)-th order zero of \(\zeta\), but for the Prime Number Theorem, we are only concerned with the area \(\Re(s) \geq 1\), where \(\zeta\) does not have any zeros.
Taken together, this means that \( f(s) \) is analytic for \( \Re(s) \geq 1 \) except for a double pole at \( s = 1 \), which we will take care of later.

context
defines \( A :: \text{nat} \Rightarrow \text{complex} \Rightarrow \text{complex} \) and \( B :: \text{nat} \Rightarrow \text{complex} \Rightarrow \text{complex} \)
defines \( A \equiv (\lambda p s. (s - 1) * \text{pre-zeta (real p)} s - \text{of-nat p} / (\text{of-nat p powr s} * (\text{of-nat p powr s} - 1))) \)
defines \( B \equiv (\lambda p s. \text{of-real (ln (real p))} / \text{of-nat p} * A p s) \)

begin
definition \( \text{pre-newman :: complex} \Rightarrow \text{complex} \) where
\( \text{pre-newman s} = (\sum p. \text{if prime p then B p s else 0}) \)
definition \( \text{newman where newman s} = 1 / (s - 1) * (\text{pre-newman s} - \text{deriv zeta s} / \text{zeta s}) \)
The sum used in the definition of \( \text{pre-newman} \) converges uniformly on any disc within the half-space with \( \Re(s) > \frac{1}{2} \) by the Weierstrass M test.

lemma \( \text{uniform-limit-pre-newman:} \)
assumes \( r :: r \geq 0 \text{ Re s} - r > 1 / 2 \)
shows \( \text{uniform-limit (cball s r)} (\lambda n s. \sum p < n. \text{if prime p then B p s else 0}) \text{ pre-newman at-top} \)
proof
from \( r \) have \( \text{Re: Re z > 1 / 2} \text{ if dist s z} \leq r \) for \( z \)
using \( \text{abs-Re-le-cmod [of s - z]} r \) that
by \( (\text{auto simp: dist-norm abs-if split: if-splits}) \)
define \( x \) where \( x = \text{Re s} - r \) — The lower bound for the real part in the disc
from \( r \text{ Re have} x > 1 / 2 \) by \( (\text{auto simp: x-def}) \)

— The following sequence \( M \) bounds the summand, and it is obviously \( O(n^{-1-\epsilon}) \) and therefore summable
define \( C \) where \( C = (\text{norm s} + r + 1) * (\text{norm s} + r) / x \)
define \( M \) where \( M = (\lambda p::\text{nat}. \text{ln p} * (C / p powr (x + 1) + 1 / (p powr x * (p powr x - 1)))) \)
show \( ?\text{thesis unfolding pre-newman-def} \)
proof \( (\text{intro Weierstrass-m-test-ev[OF eventually-mono][OF eventually-gt-at-top[of I]] ballI}) \)
show \( \text{summable M} \)
proof \( (\text{rule summable-comparison-test-bigo}) \)
define \( \varepsilon \) where \( \varepsilon = \text{min} (2 * x - 1) x / 2 \)
from \( x > 1 / 2 \) have \( \varepsilon : \varepsilon > 0 \text{ 1 +} \varepsilon < 2 * x 1 + \varepsilon < x + 1 \)
by \( (\text{auto simp: } \varepsilon-\text{def min-def field-simps}) \)
show \( M \in O(\text{Ln. powr (-1 -} \varepsilon)) \) unfolding \( \text{M-def distrib-left} \)
by \( (\text{intro sum-in-bigo}) (\text{use } \varepsilon \text{ in real-asymp})+ \)
from \( \varepsilon \) show \( \text{summable (Ln. norm (n powr (-1 -} \varepsilon))) \)
by \( (\text{simp add: summable-real-powr-iff}) \)
qed
next

88
fix  \( p : \text{nat} \) and \( z \) assume \( p : p > 1 \) and \( z : z \in \text{cball} \ s \ r \\
\)from \( z \in \text{rRe} [of \ z] \) have \( x : \text{Re} \ z \geq z \cdot x > 1 / 2 \) and \( \text{Re} \ z > 1 / 2 \\
using \text{abs}-\text{Re}-\text{le-cmod} [of \ s - z] \) by (auto simp: x-def algebra-simps dist-norm) \\
have norm-z: \( \text{norm} \ z \leq \text{norm} \ s + r \\
using \text{z-norm-triangle-ineq2} [of \ s \ r] \) by (auto simp: dist-norm norm-minus-commute) \\
from \( p > 1 \) and \( x \) and \( r \) have \( M \ p \geq 0 \\
by (auto simp: C-def M-def intro!: mult-nonneg-nonneg add-nonneg-nonneg divide-nonneg-pos) \\
have bound: \( \text{norm} \ ((z - 1) * \text{pre-zeta} \ p \ z) \leq \\
\text{norm} \ (z - 1) * (\text{norm} \ z / (\text{Re} \ z * p \ \text{powr} \ Re \ z)) \\
using \text{pre-zeta-bound} [of \ z \ p] \ p \ (\text{Re} \ z > 1 / 2), \\
\text{unfolding} \ (\text{norm-mult} \ \text{by} \ (\text{intro} \ \text{mult-mono} \ \text{pre-zeta-bound}) \ \text{auto}) \\
\text{have} \ \text{norm} \ (B \ p \ z) = \ln p / p * \text{norm} \ (A \ p \ z) \\
\text{unfolding} \ B-def \ \text{using} \ \ (p > 1) \ \text{by} \ (\text{simp add: B-def norm-mult norm-divide}) \\
\text{also have} \ \ldots \ \leq \ln p / p * (\text{norm} \ (z - 1) * \text{norm} \ z / \text{Re} \ z / p \ \text{powr} \ Re \ z + \\
p / (p \ \text{powr} \ Re \ z * (p \ \text{powr} \ Re \ z - 1))) \\
\text{unfolding} \ A-def \ \text{using} \ (p > 1) \ \text{and} \ \ldots \ \text{bound} \\
\text{by} \ (\text{intro} \ \text{mult-left-mono} \ \text{order-trans}[\text{OF norm-triangle-ineq4 add-mono} \ \text{mult-left-mono}]) \\
(\text{auto simp: norm-mult norm-mult norm-powr-real-powr} \\
\text{intro!: divide-left-mono} \ \text{order-trans}[\text{OF - norm-triangle-ineq2}]) \\
\text{also have} \ \ldots \ = \ln p * (\text{norm} \ (z - 1) * \text{norm} \ z / \text{Re} \ z / p \ \text{powr} \ (Re \ z + 1)) + \\
1 / (p \ \text{powr} \ Re \ z * (p \ \text{powr} \ Re \ z - 1))) \\
\text{using} \ (p > 1) \ \text{by} \ (\text{simp add: field-simps powr-add powr-minus}) \\
\text{also have} \ \text{norm} \ (z - 1) * \text{norm} \ z / \text{Re} \ z / p \ \text{powr} \ (Re \ z + 1) \leq C / p \ \text{powr} \\
(x + 1) \\
\text{unfolding} \ C-def \ \text{using} \ r \ (\text{Re} \ z > 1 / 2) \ \text{norm-z p x} \\
\text{by} \ (\text{intro} \ \text{mult-mono} \ \text{frac-le powr-mono order-trans}[\text{OF norm-triangle-ineq4}]) \\
\text{auto} \\
\text{also have} \ 1 / (p \ \text{powr} \ Re \ z * (p \ \text{powr} \ Re \ z - 1)) \leq \\
1 / (p \ \text{powr} \ x * (p \ \text{powr} \ x - 1)) \ \text{using} \ (p > 1) \ x \\
\text{by} \ (\text{intro} \ \text{divide-left-mono} \ \text{mult-mono powr-mono diff-right-mono mult-pos-pos}) \\
(\text{auto simp: ge-one-powr-ge-zero}) \\
\text{finally have} \ \text{norm} \ (B \ p \ z) \leq M \ p \\
\text{using} \ (p > 1) \ \text{by} \ (\text{simp add: mult-left-mono M-def}) \\
\text{with} \ (M \ p \geq 0) \ \text{show} \ \text{norm} \ (\text{if prime} \ p \ \text{then} \ B \ p \ z \ \text{else} \ 0) \leq M \ p \ \text{by simp} \\
\text{qed} \\
\text{qed} \\
\text{lemma} \ \text{suns-pre-newman}: \ \text{Re} \ s > 1 / 2 \ \text{implies} \ (\lambda p. \ \text{if prime} \ p \ \text{then} \ B \ p \ s \ \text{else} \ 0) \\
\text{suns \ pre-newman} \ s \\
\text{using} \ \text{tends-to-uniform-limit}[\text{OF uniform-limit-pre-newman}[\text{of} \ s \ 0]] \ \text{by} \ (\text{auto simp: sums-def}) \\
\text{lemma} \ \text{analytic-pre-newman} \ [\text{THEN analytic-on-subset}, \ \text{analytic-intros}]: \\
\text{pre-newman analytic-on} \ \{s. \ \text{Re} \ s > 1 / 2\} \\
\text{proof} --
have holo: (λs::complex. if prime p then B p s else 0) holomorphic-on X

if X ⊆ {s. Re s > 1 / 2} for X and p :: nat using that
  by (cases prime p)
    (auto intro!: holomorphic-intros simp: B-def A-def dest!: prime-gt-1-nat)

have holo': pre-newman holomorphic-on ball s r if r: r ≥ 0 Re s - r > 1 / 2

for s r
  proof
    from r have Re: Re z > 1 / 2 if dist s z ≤ r for z
    using abs-Re-le-cmod[of s - z] r that by (auto simp: dist-norm abs-if split: if-splits)
    show ?thesis
    by (rule holomorphic-uniform-limit[OF uniform-limit-pre-newman[of r s]])

qed

show ?thesis unfolding analytic-on-def
  proof safe
    fix s assume Re: Re s > 1 / 2
    thus ∃ r>0. pre-newman holomorphic-on ball s r
      by (intro exI[of - (Re s - 1 / 2) / 2] conjI holo') (auto simp: field-simps)

qed

qed

lemma holomorphic-pre-newman [holomorphic-intros]:
  X ⊆ {s. Re s > 1 / 2} ⇒ pre-newman holomorphic-on X
  using analytic-pre-newman by (rule analytic-imp-holomorphic)

lemma eval-fds-newman:
  assumes s: Re s > 1
  shows eval-fds fds-newman s = newman s
  proof
    have eq: (ln (real p) / real p) * hurwitz-zeta p s =
      1 / (s - 1) * (ln (real p) / (p powr (s - 1)) + B p s)
      if p: prime p for p
    proof
      have (ln (real p) / real p) * hurwitz-zeta p s =
        ln (real p) / real p * (p powr (1 - s) / (s - 1) + pre-zeta p s)
        using s by (auto simp add: hurwitz-zeta-def)
      also have ... = 1 / (s - 1) * (ln (real p) / (p powr s - 1) + B p s)
        using p s by (simp add: divide-simps powr-diff B-def)
      finally show ?thesis .
    qed

    have (λp. (ln (real p) / real p) * hurwitz-zeta p s) abs-summable-on {p. prime p}
      using s by (intro eval-fds-newman-conv-infsetsum)
    hence (λp. 1 / (s - 1) * (ln (real p) / (p powr s - 1) + B p s))
      abs-summable-on {p. prime p}
Next, we shall attempt to get rid of the pole by subtracting suitable multiples of $\zeta(s)$ and $\zeta'(s)$. To this end, we shall first prove the following alternative definition of $\zeta'(s)$:

**Lemma** deriv-zeta-eq':

**Assumes** $0 < \Re s \neq 1$

**Shows** deriv $\zeta s = \text{deriv} (\lambda z. \text{pre-zeta 1} z * (z - 1)) / (s - 1) - (\text{pre-zeta 1} s * (s - 1) + 1) / (s - 1)^2$

**Proof** (rule DERIV-imp-deriv)

**Have** [derivative-intros]: $\text{pre-zeta 1} \text{ has-field-derivative} \text{ deriv} (\text{pre-zeta 1} s)$ (at $s$)

(by (intro holomorphic-deriv[OF - UNIV] holomorphic-intros) auto)

**Have** $*:\text{deriv} (\lambda z. \text{pre-zeta 1} z * (z - 1)) s = \text{deriv} (\text{pre-zeta 1} s) s * (s - 1) + \text{pre-zeta 1} s$

(by (subst deriv-mult)

(auto intro: holomorphic-on-imp-differentiable-at[OF - UNIV] holomorphic-intros)

**Hence** $(\lambda s. \text{pre-zeta 1} s + 1 / (s - 1)) \text{ has-field-derivative} \\
\text{deriv} (\text{pre-zeta 1} s - 1 / ((s - 1) * (s - 1)))$ (at $s$)

**Using** assms by (auto intro: derivative-eq-intros)

**Also Have** deriv $(\text{pre-zeta 1} s - 1 / ((s - 1) * (s - 1))) = \text{rhs}
From this, it follows that \((s-1)ζ' (s) - ζ' (s) / ζ(s)\) is analytic for \(\Re(s) \geq 1\):

**Lemma** analytic-zeta-derivdiff:

obtains \(a\) where

\[(λz. if z = 1 then a else (z - 1) * deriv zeta z - deriv zeta z / zeta z)\]

analytic-on \(\{s. Re s ≥ 1\}\)

**Proof**

have neq: pre-zeta 1 z * (z - 1) + 1 ≠ 0 if Re z ≥ 1 for z

using zeta-Re-ge-1-nonzero[of z] that

by (cases z = 1) (auto simp: zeta-def hurwitz-zeta-def divide-simps)

let \(?y = λz. (1 - inverse (pre-zeta 1 z * (z - 1) + 1)) * ((z - 1) *

deriv ((λa. pre-zeta 1 a * (a - 1))) z - (pre-zeta 1 z * (z - 1) + 1))\)

show (λz. if z = 1 then deriv ?g 1 else (z - 1) * deriv zeta z - deriv zeta z / zeta z)

analytic-on \(\{s. Re s ≥ 1\}\) (is \(\text{if analytic-on -}\))

**Proof** (rule pole-theorem-analytic-0)

show ?g analytic-on \(\{s. 1 ≤ Re s\}\) using neq

by (auto intro!: analytic-intros)

next

show \(∃ d > 0. \forall w ∈ ball z d - \{1\}\). ?g w = (w - 1) * \(\text{if}\) w

if \(z. z ∈ \{s. 1 ≤ Re s\}\) for z

**Proof**

have \(*: \text{isCont } (λz. pre-zeta 1 z * (z - 1) + 1) z\)

by (auto intro!: continuous-intros)

obtain \(e\) where \(e > 0\) and \(e: \forall y. \text{dist } z v < e \implies \text{pre-zeta } (Suc 0) y * \)

\((y-1) + 1 ≠ 0\)

using continuous-at-avoid \([OF \text{ neq}[of z]]\) z by auto

show \(?thesis\)

**Proof** (intro exI ballI conjI)

fix \(w\)

assume \(w. w ∈ ball z (\text{min } e 1) - \{1\}\)

then have \(Re w > 0\)

using complex-Re-le-cmod [of \(z-w\)] z by (simp add: dist-norm)

with \(w\) show ?g w = (w - 1) * (if w = 1 then deriv ?g 1 else

\((w - 1) * deriv zeta w - deriv zeta w / zeta w)\)

by (subst (1 2) deriv-zeta-eq',

simp-all add: zeta-def hurwitz-zeta-def divide-simps e power2-eq-square)

(auto simp: algebra-simps)?

**QED** (use \(\text{of } e > 0\)) in auto

**QED**

auto
Finally, \( f(s) + \zeta'(s) + c\zeta(s) \) is analytic.

**lemma analytic-newman-variant:**

obtains \( c \) a where

\[
(\lambda z. \text{ if } z = 1 \text{ then } a \text{ else } \text{newman } z + \text{deriv } \zeta z + c \ast \zeta z) \text{ analytic-on } \{ s. \text{Re } s \geq 1 \}
\]

**proof**

obtain \( c \) where

\[
(\lambda z. \text{ if } z = 1 \text{ then } c \text{ else } (z - 1) \ast \text{deriv } \zeta z - \text{deriv } \zeta z / \zeta z) \text{ analytic-on } \{ s. \text{Re } s \geq 1 \}
\]

using analytic-zeta-derivdiff by blast

let \( ?g = (\lambda z. \text{ newman } z + \text{if } z = 1 \text{ then } c \text{ else } (z - 1) \ast \text{deriv } \zeta z - \text{deriv } \zeta z / \zeta z) \ast (c + \text{pre-newman } 1) \ast (\text{pre-zeta } 1 z \ast (z - 1) + 1) \)

have \( (\lambda z. \text{ if } z = 1 \text{ then } \text{deriv } ?g 1 \text{ else } \text{newman } z + \text{deriv } \zeta z + (-c + \text{pre-newman } 1)) \ast \zeta z \)

analytic-on \( \{ s. \text{Re } s \geq 1 \} \) (is \( ?f \) analytic-on -)

**proof** (rule pole-theorem-analytic-0)

show \( ?g \) analytic-on \( \{ s. 1 \leq \text{Re } s \} \)

by (intro c analytic-intros) auto

next

show \( \exists \exists d > 0. \forall w \in \text{ball } z d - \{ 1 \}. ?g w = (w - 1) \ast ?f w \)

if \( z \in \{ s. 1 \leq \text{Re } s \} \) for \( z \) using that

by (intro exI[of - 1], simp-all add: newman-def divide-simps zeta-def hurwitz-zeta-def)

(auto simp: field-simps)?

qed auto

with that show \( ?\text{thesis} \) by blast

qed

4.2 The asymptotic expansion of \( \mathfrak{M} \)

Our next goal is to show the key result that \( \mathfrak{M}(x) = \ln n + c + o(1) \).

As a first step, we invoke Ingham’s Tauberian theorem on the function we have just defined and obtain that the sum

\[
\sum_{n=1}^{\infty} \frac{\mathfrak{M}(n) - \ln n + c}{n}
\]

exists.

**lemma mertens-summable:**

obtains \( c :: \text{real} \) where summable \((\lambda n. (\mathfrak{M} n - \ln n + c) / n)\)

**proof**

from analytic-newman-variant obtain \( c \) a where
analytic: \((\lambda z. \text{ if } z = 1 \text{ then } a \text{ else } \text{newman } z + \text{deriv zeta } z + c \ast \text{zeta } z)\)

define \(f\) where \(f = (\lambda z. \text{ if } z = 1 \text{ then } a \text{ else } \text{newman } z + \text{deriv zeta } z + c \ast \text{zeta } z)\)

have analytic: \(f\) analytic-on \(\{s. \text{Re} s \geq 1\}\).

define \(F\) where \(F = \text{fds-newman } + \text{fds-deriv fds-zeta } + \text{fds-const } c \ast \text{fds-zeta}\)

note \(\text{le} = \text{conv-abscissa-add-le!}\) \(\text{conv-abscissa-deriv-le}\) \(\text{conv-abscissa-newman}\) \(\text{conv-abscissa-mult-const-left}\)

note \(\text{intros} = \text{le}[\text{THEN} \text{le-less-trans}]\) \(\text{le}[\text{THEN} \text{order-trans}]\) \(\text{fds-converges}\)

have \(\text{eval-F} = \text{eval-fds } F s = f s\) if \(s. \text{Re} s > 1\) for \(s\)

proof

have \(\text{eval-fds} F s = \text{eval-fds } (\text{fds-newman } + \text{fds-deriv fds-zeta}) s + \text{eval-fds } (\text{fds-const } c \ast \text{fds-zeta}) s\)

unfolding \(\text{F-def using} s\) by (subt eval-fds-add) (auto intro!: intros)

also have \(\ldots = f s\) using \(s\) unfolding \(\text{f-def}\)

by (subt eval-fds-add)

(auto intro!: intros simp)

finally show \(?\text{thesis}\).

qed

have \(\text{conv: fds-converges} F s\) if \(\text{Re} s \geq 1\) for \(s\)

proof (rule Newman-Ingham-1)

have \((\lambda n. \Re(\text{real } n) - \ln(\text{real } n)) \in O(\lambda c. 1)\)

using mertens-bounded by (rule landau-o.big.compose) \text{real-asympt}\n
from natfun-bigo-1E[\text{OF} this, of 1]

obtain \(c'\) where \(c': c' \geq 1 \forall n. (\Re(\text{real } n) - \ln(\text{real } n)) \leq c'\) by auto

have \(\text{Bseq } (\text{fds-nth } F)\)

proof (intro \(\text{Bseq}\) all)

fix \(n :: \text{nat}\)

show \(\text{norm } (\text{fds-nth } F n) \leq (c' + \text{norm } c)\) unfolding \(\text{F-def using}\) \(c'\)

by (auto simp: \(\text{fds-nth-zeta}\) \(\text{fds-nth-deriv}\) \(\text{fds-nth-newman}\) \(\text{scaleR-conv-of-real}\)

in-\(\text{Reals-norm}\)

intro!: order.trans[\text{OF} \text{norm-triangle-ineq}] \text{add-mono}

qed (insert \(c'\), auto intro!: add-pos-nonneg)

thus \(\text{fds-nth } F \in O(\lambda c. 1)\) by (simp add: natfun-bigo-iff-Bseq)

next

show \(f\) analytic-on \(\{s. \text{Re} s \geq 1\}\) by \(\text{fact}\)

next

show \(\text{eval-fds} F s = f s\) if \(\text{Re} s > 1\) for \(s\) using \(\text{that by}\) (rule \(\text{eval-F}\))

qed (insert \(\text{that, auto simp: F-def intro!: intros}\)

from \(\text{conv[of 1]}\) have \(\text{summable } (\lambda n. \text{fds-nth } F n) / \text{of-nat}\)

unfolding \(\text{fds-converges-def by auto}\)

also have \(\text{this} \leftrightarrow \text{summable } (\lambda n. (\Re(\text{real } n) - \ln(\text{of-nat } n)) + \Re c) / n)\)

by (intro \text{summable-cong eventually-mono} [\text{OF eventually-gt-at-top[of 0]})]

(auto simp: F-def \(\text{fds-nth-newman}\) \(\text{fds-nth-deriv}\) \(\text{fds-nth-zeta}\) \(\text{scaleR-conv-of-real}\)

intro!: \text{sum.cong dest: prime-gt-0-nat})

finally have \(\text{summable } (\lambda n. (\Re(\text{real } n) - \Re(\text{Ln}(\text{of-nat } n)) + \Re c) / n)\)

by (auto dest: \text{summable-Re})

94
also have \( \frac{\ln n}{n} \) sum goes to zero-lemma

by (intro \text{summable-cong eventually-mono}[\text{OF eventually-gt-at-top[of \(0\)]}]) (auto intro: \text{sum.cong})

finally show \( \text{thesis using that[of Re \(c\)]} \) by blast

qed

Next, we prove a lemma given by Newman stating that if the sum \( \sum a_n/n \) exists and \( a_n + \ln n \) is nondecreasing, then \( a_n \) must tend to 0. Unfortunately, the proof is rather tedious, but so is the paper version by Newman.

\textbf{lemma} sum-goestozero-lemma:

\textbf{fixes} \( d::\text{real} \)

\textbf{assumes} \( d: \{ \sum i = M\ldots N. \ a \ i / i \ < d \ \text{and le: } \forall n. \ a \ n + \ln n \leq a \ (\text{Suc } n) + \ln (\text{Suc } n) \)

and \( 0 < M \ M < N \)

\textbf{shows} \( a \ M \leq d * N / (\text{real } N - \text{real } M) + (\text{real } N - \text{real } M) / M \wedge -a \ N \leq d * N / (\text{real } N - \text{real } M) + (\text{real } N - \text{real } M) / M \)

\textbf{proof} –

have \( 0 \leq d \)

using assms by linarith+

then have \( 0 \leq d * N / (N - M + 1) \) by simp

then have le-dN: \([0 \leq x \implies x \leq d * N / (N - M + 1)] \implies x \leq d * N / (N - M + 1)\) for \( x::\text{real} \)

by linarith

have le-a-ln: \( a \ M + \ln \ M \leq a \ n + \ln n \) if \( n \geq m \) for \( n \ m \)

by (rule transitive-stepwise-le) (use le that in auto)

have \( *: x \leq b \wedge y \leq b \) if \( a \leq b \ x \leq a \ y \leq a \) for \( a \ b \ x \ y::\text{real} \)

using that by linarith

show \( \text{thesis} \)

\textbf{proof} (rule \( * \))

show \( d * N / (N - M) + \ln (N / M) \leq d * N / (\text{real } N - \text{real } M) + (\text{real } N - \text{real } M) / M \)

using \( \langle 0 < M, \ M < N, \ \ln\text{-le minus one [of } N / M]\rangle \)

by (simp add: ofNat-diff) (simp add: divide-simps)

next

have \( a \ M - \ln (N / M) \leq (d * N) / (N - M + 1) \)

\textbf{proof} (rule le-dN)

assume \( 0 \leq a \ M - \ln (N / M) \)

have \( (\text{Suc } N - M) * (a \ M - \ln (N / M)) / N = (\sum i = M\ldots N. \ (a \ M - \ln (N / M))) / N \)

by simp

also have \( \ldots \leq (\sum i = M\ldots N. \ a \ i / i) \)

\textbf{proof} (rule \text{sum-mono})

fix \( i \)

assume \( \langle 0 < M \rangle \) have \( 0 < i \) by auto

have \( (a \ M - \ln (N / M)) / N \leq (a \ M - \ln (N / M)) / i \)

using \( 0 \) using \( i : 0 < M \) by (simp add: frac-le-eq divide-simps mult-left-mono)

also have \( a \ M + \ln (\text{real } M) \leq a \ i + \ln (\text{real } N) \)

by (rule \text{order.trans[OF \text{le-a-ln[of M i]}])} (use \( i \) \text{assms in auto} 

95
hence \((a \cdot M - \ln (N / M)) / i \leq a \cdot i / \text{real } i\)

using `assms i` by (intro divide-right-mono) (auto simp: ln-div field-simps)
finally show \((a \cdot M - \ln (N / M)) / \text{real } N \leq a \cdot i / \text{real } i\).

deq

finally have \(((\operatorname{Suc }N) - M) \cdot (a \cdot M - \ln (N / M)) / N \leq \sum i = M \cdot N \cdot a \cdot i / i\)

by simp
also have \(\ldots \leq d\) using `d` by simp
finally have \(((\operatorname{Suc }N) - M) \cdot (a \cdot M - \ln (N / M)) / N \leq d\).

then show "thesis"
using \(M < N\) by (simp add: of-nat-diff field-simps)
deq

also have \(\ldots \leq d \cdot N / (N - M)\)
using `assms(1,4)` by (simp add: field-simps)
finally show \(a \cdot M \leq d \cdot N / (N - M) + \ln (N / M)\) by simp

next

have \(- a \cdot N - \ln (N / M) \leq (d \cdot N) / (N - M + 1)\)

proof (rule le-dN)

assume \(0 \leq - a \cdot N - \ln (N / M)\)

have \((\sum i = M \cdot N \cdot a \cdot i / i) \leq (\sum i = M \cdot N \cdot (a \cdot N + \ln (N / M)) / N)\)

proof (rule sum-mono)

fix `i`

assume \(i: i \in \{M \cdot N\}\)

with \(\langle 0 < M \rangle\) have \(0 < i\) by auto

have \(a \cdot i + \ln (\text{real } M) \leq a \cdot N + \ln (\text{real } N)\)

by (rule order.trans[OF \(- \ln(\text{of-nat}(\text{Suc }N))]\)) (use `assms` in auto)

hence \(a \cdot i / i \leq (a \cdot N + \ln (N / M)) / i\)

using `assms(3,4)` by (intro divide-right-mono) (auto simp: field-simps)

\(\ln\text{-die}\)

also have \(\ldots \leq (a \cdot N + \ln (N / M)) / N\)

using \(\langle i \cdot i > 0\rangle\) by (intro divide-left-mono-neg) auto

finally show \(a \cdot i / i \leq (a \cdot N + \ln (N / M)) / N\).
deq

also have \(\ldots = ((\operatorname{Suc }N) - M) \cdot (a \cdot N + \ln (N / M)) / N\)

by simp

finally have \((\sum i = M \cdot N \cdot a \cdot i / i) \leq (\ln(\operatorname{Suc }N) - \ln(\text{real } M)) \cdot (a \cdot N + \ln (N / M)) / N\)

using \(\langle M < N\rangle\) by (simp add: of-nat-diff)

then have \(-((\ln(\operatorname{Suc }N) - \ln(\text{real } M)) \cdot (a \cdot N + \ln (N / M)) / N) \leq |\sum i|

= M \cdot N \cdot a \cdot i / i\)

by linarith

also have \(\ldots \leq d\) using `d` by simp

finally have \(-((\ln(\operatorname{Suc }N) - \ln(\text{real } M)) \cdot (a \cdot N + \ln (N / M)) / N) \leq d\).

then show "thesis"
using \(\langle M < N\rangle\) by (simp add: of-nat-diff field-simps)
deq

also have \(\ldots \leq d \cdot N / \text{real } (N - M)\)

using \(\langle 0 < M \cdot \langle M < N\rangle\times 0 < d\rangle\) by (simp add: field-simps)

finally show \(-a \cdot N \leq d \cdot N / \text{real } (N - M) + \ln (N / M)\) by simp

96
proposition sum-goestozero-theorem:
assumes summ: summable (λi. a i / i)
and le: \( \forall n. a n + \ln n \leq a (\text{Suc } n) + \ln (\text{Suc } n) \)
shows \( a \xrightarrow{\text{a}} 0 \)
proof (clarsimp simp: lim-sequentially)
fix r::real
assume r > 0
have *: \( \exists n0. \forall n \geq n0. |a n| < \varepsilon \) if \( \varepsilon: 0 < \varepsilon \varepsilon < 1 \) for \( \varepsilon \)
proof –
  have 0 < (\varepsilon / 8)^2 using 0 < \varepsilon by simp
  then obtain N0 where N0: \( \forall m n. m \geq N0 \implies \text{norm } \sum_{k=m..n} (\lambda i. a i) \frac{1}{k} < (\varepsilon / 8)^2 \)
by (metis summable-partial-sum-bound summ)
  obtain N1 where \( \text{real } N1 > 4 / \varepsilon \) by auto
  hence N1 ≠ 0 and N1: 1 / real N1 < \varepsilon / 4 using \( \varepsilon \)
  by (auto simp: divide-simps mult-ac intro: Nat.gr0I)
  have |a n| < \varepsilon if n: n ≥ 2 * N0 + N1 + 7 for n
  proof –
    define k where k = [n * \varepsilon / 4]
    have n * \varepsilon / 4 > 1 and n * \varepsilon / 4 ≤ n / 4 and n / 4 < n
    using less-le-trans[OF N1, of n / N1 * \varepsilon / 4] \( \varepsilon \) by auto
    hence k: k > 0 \( \frac{4 \ast k \leq n \text{ nat } k < n \ast \varepsilon / 4} \) – 1 < k k ≤ (n * \varepsilon / 4)
    unfolding k-def by linarith+
  have –a n < \varepsilon
  proof –
    have N0 ≤ n – nat k
    using n k by linarith
    then have *: \( \sum k = n – nat k .. n. a k / k | < (\varepsilon / 8)^2 \)
    using N0 [of n – nat k n] by simp
    have –a n ≤ (\varepsilon / 8)^2 * n / (\varepsilon * \varepsilon / 4) + |n * \varepsilon / 4| / (n – k)
    using sum-goestozero-lemma [OF * le, THEN conjunct2] k by (simp add: of-nat-diff k-def)
    also have . . . < \varepsilon
  proof –
    have \varepsilon / 16 * n / k < 2
    using k by (auto simp: field-simps)
    then have \varepsilon * (\varepsilon / 16 * n / k) < \varepsilon * 2
    using \varepsilon mult-less-cancel-left-pos by blast
    then have (\varepsilon / 8)^2 * n / k < \varepsilon / 2
    by (simp add: field-simps power2-eq-square)
    moreover have k / (n – k) < \varepsilon / 2
    proof –
have \((\varepsilon + 2) \cdot k < 4 \cdot k\) using \(k \varepsilon\) by simp

also have ... \(\leq \varepsilon \cdot \text{real } n\) using \(k\) by (auto simp: field-simps)

finally show \(?\text{thesis}\) using \(k\) by (auto simp: field-simps)

qed

ultimately show \(?\text{thesis}\) unfolding \(k\)-def by linarith

qed

finally show \(?\text{thesis}\).

qed

moreover have \(a \cdot n < \varepsilon\)

proof –

have \(N0 \leq n\) using \(n k\) by linarith

then have \(*: \sum k = n \cdot n + \text{nat } k\cdot a k / k < (\varepsilon / 8)^2\)

using \(N0\) [of \(n\) \(n + \text{nat } k\)] by simp

have \(a \cdot n \leq (\varepsilon / 8)^2 \cdot (n + \text{nat } k) / k + k / n\)

using \text{sum-goestozero-lemma} [\(\text{OF } \text{le}, \text{THEN conjunct1}\)] \(k\) by (simp add: of-nat-diff)

also have ... \(< \varepsilon\)

proof –

have \(4 \leq 28 \cdot \text{real-of-int } k\) using \(k\) by linarith

then have \(\varepsilon / 16 \cdot n / k < 2\) using \(k\) by (auto simp: field-simps)

have \(\varepsilon \cdot (\text{real } n + k) < 32 \cdot k\)

proof –

have \(\varepsilon \cdot n / 4 < k + 1\) by (simp add: mult.commute \(k\)-def)

then have \(\varepsilon \cdot n < 4 \cdot k + 4\) by (simp add: divide-simps)

also have ... \(\leq 8 \cdot k\) using \(k\) by auto

finally have \(1: \varepsilon \cdot \text{real } n < 8 \cdot k\).

have \(2: \varepsilon \cdot k < k\) using \(k \varepsilon\) by simp

show \(?\text{thesis}\) using \(k\) \text{add-strict-mono} [\(\text{OF } 1 \ 2\)] by (simp add: algebra-simps)

qed

ultimately show \(?\text{thesis}\).

qed

ultimately show \(?\text{thesis}\) by force

qed

then show \(?\text{thesis}\) by blast

qed

show \(\exists n0. \ \forall n \geq n0. \ |a \cdot n| < r\)

using \(\text{of } \text{min } r\) \((1/5)\) \(\langle 0 < r\rangle\) by force

qed

This leads us to the main intermediate result:

lemma \text{Mertens-convergent}: \text{convergent} \((\lambda n::\text{nat}. \ \Re \ n - \ln n)\)

proof –


obtain $c$ where $c\text{: summable}\ (\lambda n. (\Re n - \ln n + c) / n)$
  by (blast intro: mertens-summable)
then obtain $l$ where $l\text{:} (\lambda n. (\Re n - \ln n + c) / n) \text{ sums } l$
  by (auto simp: summable-def)

have $\ast\text{:} (\lambda n. \Re n - \ln n + c) \to 0$
  by (rule sum-goestozero-theorem[OF $c$]) auto

hence $\lambda n. \Re n - \ln n \to -c$
  by (simp add: tendsto-iff dist-norm)
thus $}\text{ thesis by (rule convergentI)}$
qed

corollary $\Re\text{-minus-ln-limit}$:
  obtains $c$ where $((\lambda x\::\text{real}. \Re x - \ln x) \to c) \text{ at-top}$
proof –
  from Mertens-convergent obtain $c$ where $\lambda n. \Re n - \ln n \to c$
  by (auto simp: convergent-def)
hence $1\text{:} ((\lambda x\::\text{real}. \Re (\text{nat} \lfloor x \rfloor) - \ln (\text{nat} \lfloor x \rfloor)) \to c) \text{ at-top}$
  by (rule filterlim-compose real-asymp)
have $2\text{:} ((\lambda x\::\text{real}. \ln (\text{nat} \lfloor x \rfloor) - \ln x) \to 0) \text{ at-top}$
  by real-asymp
have $3\text{:} ((\lambda x. \Re x - \ln x) \to c) \text{ at-top}$
  using tendsto-add[OF 1 2] by simp
with that show $\text{thesis by blast}$
qed

4.3 The asymptotics of the prime-counting functions

We will now use the above result to prove the asymptotics of the prime-
counting functions $\vartheta(x) \sim x$, $\psi(x) \sim x$, and $\pi(x) \sim x/\ln x$. The last of these
is typically called the Prime Number Theorem, but since these functions can
be expressed in terms of one another quite easily, knowing the asymptotics
of any of them immediately gives the asymptotics of the other ones.
In this sense, all of the above are equivalent formulations of the Prime Num-
ber Theorem. The one we shall tackle first, due to its strong connection to
the $\Re$ function, is $\vartheta(x) \sim x$.

We know that $\Re(x)$ has the asymptotic expansion $\Re(x) = \ln x + c + o(1)$.
We also know that

$$\vartheta(x) = x\Re(x) - \int_{2}^{x} \Re(t) \, dt.$$ 

Substituting in the above asymptotic equation, we obtain:

$$\vartheta(x) = x\ln x + cx + o(x) - \int_{2}^{x} \ln t + c + o(1) \, dt$$

$$= x\ln x + cx + o(x) - (x \ln x - x + cx + o(x))$$

$$= x + o(x)$$
In conclusion, \( \vartheta(x) \sim x \).

**Theorem \( \vartheta \)-asymptotics: \( \vartheta \sim [\text{at-top}] (\lambda x. x) \)**

**Proof**

- from \( \mathbb{R} \)-minus-ln-limit obtain \( c \) where \( c \) : \((\lambda x. \mathbb{R} x - \ln x) \longrightarrow c\) at-top
  - by auto
- define \( r \) where \( r = (\lambda x. \mathbb{R} x - \ln x - c) \)
  - by \((\text{simp add: } r\text{-def})\)
- have \( r \in o(\lambda x. 1) \) unfolding \( r\text{-def} \)
  - using \( \text{tendsto-add}[OF } c \text{ tendsto-const}[of } -c\] by \((\text{intro smalloI-tendsto})\) auto

- define \( r' \) where \( r' = (\lambda x. \text{integral } (\lambda x. 2 \cdot x) \cdot r) \)
  - by \((\text{intro integrable-diff integrable-primes-M})\)
  - (auto intro: integrable-continuous-real continuous-intros)
- hence integral: \( (r \text{ has-integral } r' \text{ x }) \) \((\lambda x. 2 \cdot x) \) if \( x \geq 2 \) for \( x \)
  - by \((\text{auto simp: has-integral-iff } r'\text{-def})\)
- have \( r' \in o(\lambda x. x) \) using \( \text{integrable-r unfolding } r'\text{-def} \)
  - by \((\text{intro integral-smallo}[OF } r\] by \((\text{auto simp: filterlim-ident})\)

- define \( C \) where \( C = 2 \cdot (c + \ln 2 - 1) \)
  - by \((\text{auto simp: field-simps } \mathbb{R}\text{-expand } C\text{-def})\)
- have \( \vartheta \sim [\text{at-top}] (\lambda x. x + (r x \cdot x + C - r' x)) \)
  - by \((\text{intro has-integral-add}[OF } \text{fundamental-theorem-of-calculus integral])\)
  - (auto simp flip: has-real-derivative-iff-has-vector-derivative)
- from \( \text{has-integral-unique}[OF } \vartheta\text{-conv-} \mathbb{R}\text{-integral this}\]
  - show \( \vartheta x = x + (r x \cdot x + C - r' x) \) using \( x \)
  - by \((\text{simp add: field-simps } \mathbb{R}\text{-expand } C\text{-def})\)
- qed

also have \( (\lambda x. r x \cdot x + C - r' x) \in o(\lambda x. x) \)

**Proof**

- show \( (\lambda x. C) \in o(\lambda x. x) \) by real-asym
- qed \((\text{insert landau-o.small-big-mult}[OF } r\] of \( \lambda x. x \) \( r'\), simp-all\)
- hence \( (\lambda x. x + (r x \cdot x + C - r' x)) \sim [\text{at-top}] (\lambda x. x) \)
  - by \((\text{subt asymp-equiv-add-right})\) auto
- finally show \( \vartheta\)-thesis by auto
- qed

The various other forms of the Prime Number Theorem follow as simple corollaries.

**Corollary \( \psi \)-asymptotics: \( \psi \sim [\text{at-top}] (\lambda x. x) \)**

- using \( \vartheta\)-asymptotics PNT4-imp-PNT5 by simp
corollary prime-number-theorem: \( \pi \sim \) \([\lambda x. x / \ln x]\)
using \( \vartheta \)-asymptotics PNT4-imp-PNT1 by simp

corollary \( \ln \pi \)-asymptotics: \( (\lambda x. \ln (\pi x)) \sim [\lambda (\lambda x. x) \to 1] \)
using prime-number-theorem PNT1-imp-PNT1' by simp

corollary \( \pi \ln \pi \)-asymptotics: \( (\lambda x. \pi x \cdot \ln (\pi x)) \sim [\lambda (\lambda x. x) \to 1] \)
using prime-number-theorem PNT1-imp-PNT2 by simp

corollary nth-prime-asymptotics: \( (\lambda n. \text{real} \ (\text{nth-prime } n)) \sim [\lambda (\lambda n. n \cdot \ln (\text{real } n)) \to 1] \)
using \( \pi \ln \pi \)-asymptotics PNT2-imp-PNT3 by simp

The following versions use a little less notation.

corollary prime-number-theorem': \( ((\lambda x. \pi x / (x / \ln x) \to 1) \) at-top
using prime-number-theorem by (rule asymp-equivD-strong[OF eventually-mono[OF eventually-gt-at-top[of 1]]]) auto

corollary prime-number-theorem'": \( (\lambda x. \text{card} \ {p. \text{prime } p \land \text{real } p \leq x}) \sim [\lambda (\lambda x. x / \ln x) \to 1] \)
proof –
have \( \pi = (\lambda x. \text{card} \ {p. \text{prime } p \land \text{real } p \leq x}) \)
by (intro ext) (simp add: \( \pi \)-def prime-sum-upto-def)
with prime-number-theorem show ?thesis by simp
qed

corollary prime-number-theorem'": \( (\lambda n. \text{card} \ {p. \text{prime } p \land \text{real } p \leq n}) \sim [\lambda (\lambda n. n \cdot \ln (\text{real } n) \to 1] \)
proof –
have \( (\lambda n. \text{card} \ {p. \text{prime } p \land \text{real } p \leq \text{real } n}) \sim [\lambda (\lambda n. n \cdot \ln (\text{real } n) \to 1] \)
using prime-number-theorem'" by (rule asymp-equiv-compose) (simp add: filterlim-real-sequentially)
thus ?thesis by simp
qed

end

5 Mertens’ Theorems

theory Mertens-Theorems
imports
Prime-Counting-Functions
Stirling-Formulas
begin

In this section, we will prove Mertens’ First and Second Theorem. These are
weaker results than the Prime Number Theorem, and we will derive them without using it.

However, like Mertens himself, we will not only prove them *asymptotically*, but *absolutely*. This means that we will show that the remainder terms are not only “Big-O” of some bound, but we will give concrete (and reasonably tight) upper and lower bounds for them that hold on the entire domain. This makes the proofs a bit more tedious.

## 5.1 Absolute Bounds for Mertens’ First Theorem

We have already shown the asymptotic form of Mertens’ first theorem, i.e. \( M(n) = \ln n + O(1) \). We now want to obtain some absolute bounds on the \( O(1) \) remainder term using a more careful derivation than before.

The precise bounds we will show are \( M(n) - \ln n \in (-1 - \frac{9}{\pi^2}; \ln 4 \approx (-1.9119; 1.3863) \) for \( n \in \mathbb{N} \).

First, we need a simple lemma on the finiteness of exponents to consider in a sum of all prime powers up to a certain point:

**lemma** exponents-le-finite:

```lean
assumes p > (1 :: nat) k > 0
shows finite {i. real (p ^ (k * i + l)) \leq x}
proof (rule finite-subset)
show {i. real (p ^ (k * i + l)) \leq x} \subseteq {...nat \{x\}}
proof safe
fix i assume i: real (p ^ (k * i + l)) \leq x
have i < 2 ^ i by (rule less-exp)
also from assms have i \leq k * i + l by (cases k) auto
hence 2 ^ i \leq (2 ^ (k * i + l)) :: nat
using assms by (intro power-increasing) auto
also have ... \leq p ^ (k * i + l) using assms by (intro power-mono) auto
also have real ... \leq x using i by simp
finally show i \leq nat \{x\} by linarith
qed
qed auto
```

Next, we need the following bound on \( \zeta'(2) \):

**lemma** deriv-zeta-2-bound: Re (deriv zeta 2) > -1

```lean
proof
have ((\lambda x::real. ln (x + 3) * (x + 3) powr -2) has-integral (ln 3 + 1) / 3) (interior {0..})
using ln-powr-has-integral-at-top[of 1 0 3 -2]
by (simp add: interior-real-atLeast powr-minus)
hence ((\lambda x::real. ln (x + 3) * (x + 3) powr -2) has-integral (ln 3 + 1) / 3) {0..}
by (subst (asm) has-integral-interior) auto
also have ?this \iff ((\lambda x::real. ln (x + 3) / (x + 3) ^ 2) has-integral (ln 3 + 1) / 3) {0..}
```

102
by (intro has-integral-cong) (auto simp: powr-minus field-simps)

finally have int: . . .
have \( \exp (1/2 :: \text{real}) \leq 2^2 \)
  using \( \exp \leq \) (subst \( \exp \cdot \text{double} \) [symmetric]) simp-all
hence \( \exp \cdot \text{half} \): \( \exp (1/2 :: \text{real}) \leq 2 \)
  by (rule power2-le-imp-le) auto

have \( \text{mono} \): \( \ln x / x^2 \leq \ln y / y^2 \) \( \text{if } y \geq \exp (1/2) x \geq y \) \( \text{for } x y :: \text{real} \)

proof (rule DERIV-nonpos-imp-nonincreasing[of \( - \cdot \lambda x. \ln x / x^2 \)])
  fix \( t \) assume \( t \geq y t \leq x \)
  have \( y > 0 \) by (rule less-le-trans[of \( - \cdot \text{that}(1) \)]) auto
  with \( t \) that \( \ln t \geq \ln (\exp (1/2)) \)
  by (subst ln-le-cancel-if) auto
hence \( \ln t \geq 1/2 \) by (simp only: ln-exp)

from \( t \cdot y > 0 \) have \( ((\lambda x. \ln x / x^2) \text{ has-field-derivative } ((1 - 2 * \ln t) / t^3)) \) (\( \text{at } t \))
  by (auto intro!: derivative-eq-intros simp: eval-nat-numeral field-simps)

moreover have \( (1 - 2 * \ln t) / t^3 \leq 0 \)
  using \( t \) that \( y > 0 \cdot \ln t \geq 1/2 \) by (intro divide-nonpos-pos) auto

ultimately show \( 3 f' \cdot ((\lambda x. \ln x / x^2) \text{ has-field-derivative } f') \) (\( \text{at } t \)) \( \wedge f' \leq 0 \) by blast

qed fact+

have \( \text{fds-converges } (\text{fds-deriv fds-zeta}) \) (\( 2 :: \text{complex} \))
  by (intro fds-converges-deriv) auto
hence \( (\lambda n. \text{of-real } (-\ln (\text{real } (\text{Suc } n)) / (\text{of-real } (\text{Suc } n)) ^ 2)) \text{ sums deriv zeta } 2 \)
  by (auto simp: fds-converges-altdef add-ac eval-fds-deriv-zeta fds-nth-deriv scaleR-conv-of-real
    simp del: of-real-Suc)

note * = sums-split-initial-segment[of sums-minus[of sums-\text{Re}[of \( \text{this} \)], of \( \text{3} \)]

have \( (\lambda n. \ln (\text{real } (n+4)) / (\text{real } (n+4) ^ 2) \text{ sums } (-\text{Re } (\text{deriv zeta } 2) - (\ln 2 / 4 + \ln 3 / 9)) \)
  using * by (simp add: eval-nat-numeral)

hence \( -\text{Re } (\text{deriv zeta } 2) - (\ln 2 / 4 + \ln 3 / 9) = \)
  \( (\sum n. \ln (\text{real } (\text{Suc } n)) + 3) / (\text{real } (\text{Suc } n) + 3) ^ 2) \)
  by (simp-all add: sums iff algebra-simps)
also have \( \ldots \leq (\ln 3 + 1) / 3 \) using int exp-half
  by (intro decreasing-sum-le-integral divide-nonneg-pos mono) (auto simp: powr-minus field-simps)

finally have \( -\text{Re } (\text{deriv zeta } 2) \leq (16 * \ln 3 + 9 * \ln 2 + 12) / 36 \)
  by simp
also have \( \ln 3 \leq (11 / 10 :: \text{real}) \)
  using ln-approx-bounds[of \( 3 \)] by (simp add: power-numeral-reduce numeral-2-eq2)

hence \( (16 * \ln 3 + 9 * \ln 2 + 12) / 36 \leq (16 * (11 / 10) + 9 * 25 / 36 + 12) / (36 :: \text{real}) \)
  using ln2-le-25-over-36 by (intro add-monol mono left-1 mono divide-right-1 mono)
  auto
also have \( \ldots < 1 \) by simp

103
finally show ?thesis by simp

qed

Using the logarithmic derivative of Euler’s product formula for $\zeta(s)$ at $s = 2$ and the bound on $\zeta'(2)$ we have just derived, we can obtain the bound

$$\sum_{p^i \leq x, i \geq 2} \frac{\ln p}{p^i} < \frac{9}{\pi^2}.$$
have finite: finite \{ p. prime p ∧ real p ≤ x \}
    by (rule finite-subset[OF - finite-Nats-le-real[of x]]) (auto dest: prime-gt-0-nat)

have finite S' unfolding S'-alt using finite-row[of - 0]
    by (intro finite-Sigma finite) auto

have R ≤ 3 / 2 * (∑(p, i) | (p, i) ∈ S' ∧ even i. ln (real p) / real (p ^ i))
proof
  have R = (∑y∈\{0, 1\}. ∑z | z ∈ S' ∧ snd z mod 2 = y. ln (real (fst z)) / real (fst z ^ snd z))
    using 'finite S' by (subst sum) (auto simp: case-prod-unfold R-def S'-def)
  also have ... = (∑(p,i) | (p, i) ∈ S' ∧ even i. ln (real p) / real (p ^ i)) + (∑(p,i) | (p, i) ∈ S' ∧ odd i. ln (real p) / real (p ^ i))
  unfolding even-iff-mod-2-eq-zero odd-iff-mod-2-eq-one by (simp add: case-prod-unfold)
  also have (∑(p,i) | (p, i) ∈ S' ∧ odd i. ln (real p) / real (p ^ i)) = (∑(p,i) | (p, i) ∈ S'' ∧ even i. ln (real p) / real (p ^ Suc i))
    by (intro sum.reindex-bij-witness[of - λ(p,i). (p, Suc i) λ(p,i). (p, i - 1)])
    (auto simp: case-prod-unfold S'-def S''-def elim: oddE simp del: power-Suc)
  also have ... ≤ (∑(p,i) | (p, i) ∈ S' ∧ even i. ln (real p) / real (p ^ Suc i))
  unfolding case-prod-unfold by (intro sum-mono2 divide-nonneg-pos ln-ge-zero finite-subset[OF - finite S''])
    (auto simp: S'-def S''-def case-prod-unfold dest: prime-gt-0-nat simp del: power-Suc)
  also have ... = (1 / 2) * (∑(p,i) | (p, i) ∈ S' ∧ even i. ln (real p) / real (p ^ i))
  unfolding case-prod-unfold by (intro sum-monono divide-left-mono) (auto simp: S'-def dest!: prime-gt-1-nat)
  also have ... = 3 / 2 * (∑(p,i) | (p, i) ∈ S' ∧ even i. ln (real p) / real (p ^ i)) + ...
    by simp
  finally show ?thesis by simp

qed
also have \[ \sum_{\text{prime } p \land \text{real } p \leq x} \ln \left( \frac{\text{real } p}{\sqrt{\text{real } p^2 - 1}} \right) \leq \left( \sum_{\text{prime } p} \ln \left( \frac{\text{real } p}{\sqrt{\text{real } p^2 - 1}} \right) \right) \]

\[ \text{proof (rule sum-mono)} \]

fix \( p \) assume \( p \in \{ p. \text{prime } p \land \text{real } p \leq x \} \)

have \( p > 1 \) using \( p \) by (auto dest: prime-gt-1-nat)

have \( \sum_{i} i \mid i > 0 \land \text{even } i \land \text{real } (p ^ i) \leq x. (1 / \text{real } p ^ i) = (\sum_{i} i \mid \text{real } (p ^ (2 * i + 2)) \leq x. (1 / \text{real } p ^ (2 * i))) / \text{real } p ^ 2 \)

(is - = ?S / - unfolding sum-divide-distrib
by (rule sum.reindex-bij-witness[of - \lambda i. 2 * Suc i \lambda. (i - 2) div 2])
(insert \( p > 1 \), auto simp: numeral-3-eq-3 power2-eq-square power-diff algebra-simps elim!: evenE)

also have \( ?S = \sum_{i} i \mid \text{real } (p ^ (2 * i + 2)) \leq x. (1 / \text{real } p ^ (2 * i)) \)

by (subst power-mult) (simp-all add: algebra-simps power-divide)

also have \( \sum_{i} i \mid \text{real } p ^ 2 / \text{real } p ^ 2 - 1 \)

using exponents-le-finite[of \( 2 \ 2 \ x \) \( p > 1 \)]
by (intro sum-le-suminf) (auto simp: summable-geometric-iff)

also have \( \sum_{i} i \mid \text{real } p ^ 2 = 1 / \text{real } p ^ 2 - 1 \)

using \( p > 1 \) by (simp add: divide-simps)

finally have \( \sum_{i} i \mid 0 < i \land \text{even } i \land \text{real } (p ^ i) \leq x. (1 / \text{real } p ^ i) \leq 1 / (\text{real } p ^ 2 - 1) \) (is ?lhs \leq ?rhs)

using \( p > 1 \) by (simp add: divide-right-mono)

thus \( \ln \left( \frac{\text{real } p}{\sqrt{\text{real } p^2 - 1}} \right) \leq \ln \left( \frac{\text{real } p}{\sqrt{\text{real } p^2 - 1}} \right) \)

using \( p > 1 \) by (simp add: divide-simps)

\( \text{qed} \)

also have \( \sum_{a} a \mid \text{prime } p \land \text{real } p \leq x. \ln \left( \frac{\text{real } p}{\sqrt{\text{real } p^2 - 1}} \right) \)

using finite by (intro infsetsum-finite [symmetric] auto)

also have \( \sum_{a} a \mid \text{prime } p. \ln \left( \frac{\text{real } p}{\sqrt{\text{real } p^2 - 1}} \right) \)

using eval-fds-logderiv-zeta-real[of \( 2 \)] finite
by (intro infsetsum-mono-neutral-left divide-nonneg-pos) (auto simp: dest: prime-gt-1-nat)

also have \( \sum_{a} a \mid \neg \text{Re } \left( \text{deriv } \zeta \text{-real } (\text{of-real } 2) \right) / \zeta \text{-real } (\text{of-real } 2) \)

by (subt eval-fds-logderiv-zeta-real auto)

also have \( \sum_{a} a \mid \neg (\text{Re } (\text{deriv } \zeta \text{-real } 2)) \ast (6 / \pi^2) \)

by (simp add: zeta-even-numeral)

also have \( \sum_{a} a < 1 \ast (6 / \pi^2) \)

using deriv-zeta-2-bound by (intro mult-strict-right-mono auto)

also have \( 3 / 2 \ast \ldots = 9 / \pi^2 \) by simp

finally show \( \text{thesis by simp} \)

\( \text{qed} \)

We now consider the equation

\[ \ln(n!) = \sum_{k \leq n} \Lambda(k) \left\lfloor \frac{n}{k} \right\rfloor \]

and estimate both sides in different ways. The left-hand-side can be estimated using Stirling’s formula, and we can simplify the right-hand side.
to
\[
\sum_{k \leq n} \Lambda(k) \left\lfloor \frac{n}{k} \right\rfloor = \sum_{p^i \leq x, i \geq 1} \ln p \left\lfloor \frac{n}{p^i} \right\rfloor
\]
and then split the sum into those \( p^i \) with \( i = 1 \) and those with \( i \geq 2 \).

Applying the bound we have just shown and some more routine estimates, we obtain the following reasonably strong version of Mertens’ First Theorem on the naturals: \( \mathfrak{M}(n) - \ln(n) \in (-1 - \frac{9}{\pi^2}; \ln 4] \)

**Theorem** mertens-bound-strong:
- **Fixes** \( n :: \text{nat} \)
- **Assumes** \( n :: n > 0 \)
- **Shows** \( \mathfrak{M} n - \ln n \in \{-1 - \frac{9}{\pi^2} < \ldots \ln 4\} \)

**Proof (cases \( n \geq 3 \))**
- **Case** False
  - **With** \( n \) consider \( n = 1 \mid n = 2 \) by force
  - **Thus** ?thesis
  - **Proof** cases
    - **Assume** [simp]: \( n = 1 \)
      - **Have** \(-1 + (\frac{-9}{\pi^2}) < 0 \)
      - **By** (intro add-neg-neg divide-neg-pos) auto
      - **Thus** ?thesis by simp
    - **Next**
      - **Assume** [simp]: \( n = 2 \)
      - **Have** eq: \( \mathfrak{M} n - \ln n = -\ln 2 / 2 \) by (simp add: eval-M)
      - **Have** \(-1 - \frac{9}{\pi^2} + \ln 2 / 2 \leq -1 - \frac{9}{4} + \frac{25}{36} / 2 \)
      - **Using** pi-less-4 ln2-le-25-over-36
      - **By** (intro diff-mono add-mono divide-left-mono divide-right-mono power-mono) auto
      - **Also have** \ldots < 0 by simp
      - **Finally have** \(-\ln 2 / 2 > -1 - \frac{9}{\pi^2} \) by simp
      - **Moreover** {
        - **Have** \(-\ln 2 / 2 \leq (0::real) \) by (intro divide-nonpos-pos) auto
        - **Also have** \ldots \leq \ln 4 by simp
        - **Finally have** \(-\ln 2 / 2 \leq \ln (4 :: \text{real}) \) by simp
      }
    - **Ultimately show** ?thesis unfolding eq by simp
  - **Qed**
- **Next**
  - **Case** True
  - **Hence** \( n :: n \geq 3 \) by simp
  - **Have** finite: finite \{ \( (p, i). \) prime \( p \wedge i \geq 1 \wedge p^i \leq n \} \)
  - **Proof** (rule finite-subset)
    - **Show** \{ \( (p, i). \) prime \( p \wedge i \geq 1 \wedge p^i \leq n \} \subseteq \{..\text{nat} \ [\text{root} 1 (\text{real} n)]\} \times \{..\text{nat} \ [\log 2 (\text{real} n)]\} \)
    - **Using** primepows-les-subset[of real n 1] \( n \) unfolding of-nat-le-iff by auto
  - **Qed** auto
- **Define** \( r \) where \( r = \text{prime-sum-upto} (\lambda p. \ln (\text{real} p) * \text{frac} (\text{real} n / \text{real} p)) n \)

107
\texttt{define} \textit{R} \texttt{where} \ R = (\sum (p,i) \mid \text{prime } p \land i > 1 \land p \sim i \leq n. \ \ln (\text{real } p) * \text{real } (n \div (p \sim i))) \\
\texttt{define} \textit{R'} \texttt{where} \ R' = (\sum (p,i) \mid \text{prime } p \land i > 1 \land p \sim i \leq n. \ \ln (\text{real } p) / p \sim i) \\
\texttt{have} \ [\text{simp}]: \ \ln (4 \div \text{real}) = 2 * \ln 2 \\
\quad \text{using} \ \ln\text{-realpow}[\text{of } 2 \ 2] \text{ by simp} \\
\texttt{from pi-less-4} \ \texttt{have} \ \ln \pi \leq \ln 4 \ \text{by} \ (\text{subst ln-le-cancel-iff}) \ \text{auto} \\
\texttt{also have} \ \ldots = 2 * \ln 2 \ \text{by simp} \\
\texttt{also have} \ \ldots \leq 2 * (25 / 36) \ \text{by} \ (\text{intro mult-left-mono} \ \ln\text{-2-25-over-36}) \ \text{auto} \\
\texttt{finally have} \ \ln\pi \ \leq 25 / 18 \ \text{by simp} \\
\texttt{have} \ \ln 3 \leq \ln (4^{\div\text{nat}}) \ \text{by} \ (\text{subst ln-le-cancel-iff}) \ \text{auto} \\
\texttt{also have} \ \ldots = 2 * \ln 2 \ \text{by simp} \\
\texttt{also have} \ \ldots \leq 2 * (25 / 36) \ \text{by} \ (\text{intro mult-left-mono} \ \ln\text{-2-25-over-36}) \ \text{auto} \\
\texttt{finally have} \ \ln\text{-3}: \ \ln (3^{\div\text{real}}) \leq 25 / 18 \ \text{by simp} \\
\texttt{have} \ R / n = (\sum (p,i) \mid \text{prime } p \land i > 1 \land p \sim i \leq n. \ \ln (\text{real } p) * (\text{real } (n \div (p \sim i)) / n)) \\
\quad \text{by} \ (\text{simp add: } R\text{-def sum-divide-distrib field-simps case-prod-unfold}) \\
\texttt{also have} \ \ldots \leq (\sum (p,i) \mid \text{prime } p \land i > 1 \land p \sim i \leq n. \ \ln (\text{real } p) * (1 / p \sim i)) \\
\texttt{unfolding} \ R'\text{-def case-prod-unfold} \ \texttt{using} \ n \\
\quad \text{by} \ (\text{intro sum-mono} \ \text{mult-left-mono}) \ (\text{auto simp: field-simps real-of-nat-div dest: prime-gt-0-nat}) \\
\texttt{also have} \ \ldots = R' \ \text{by} \ (\text{simp add: } R'\text{-def}) \\
\texttt{also have} \ R' < 9 / p^2 \\
\texttt{unfolding} \ R'\text{-def} \ \texttt{using} \ \text{mertens-remainder-aux-bound[of } n\text{]} \ \text{by simp} \\
\texttt{finally have} \ R / n < 9 / p^2 . \\
\texttt{moreover have} \ R \geq 0 \\
\texttt{unfolding} \ R\text{-def} \ \texttt{by} \ (\text{intro sum-nonneg} \ \text{mult-nonneg-nonneg}) \ (\text{auto dest: prime-gt-0-nat}) \\
\texttt{ultimately have} \ R\text{-bounds}: \ R / n \in \{0..<9 / p^2\} \ \text{by simp} \\
\texttt{have} \ \ln (\text{fact } n \div \text{real}) \leq \ln (2 * p \pi n) / 2 + n * \ln n - n + 1 / (12 * n) \\
\quad \text{using} \ \ln\text{-fact-bounds}(2)[\text{of } n] \ n \ \text{by simp} \\
\texttt{also have} \ \ldots / n - \ln n = -1 + (\ln 2 + \ln p) / (2 * n) + (\ln n / n) / 2 + 1 / (12 * n^2) \\
\quad \text{using} \ n \ \text{by} \ (\text{simp add: } \text{power2-eq-square field-simps ln-mult}) \\
\texttt{also have} \ \ldots \leq -1 + (\ln 2 + \ln p) / (2 * 3) + (\ln 3 / 3) / 2 + 1 / (12 * 3^2) \\
\quad \text{using} \ \exp-le-n p \pi g t 3 \\
\quad \text{by} \ (\text{intro add-mono} \ \text{divide-right-mono} \ \text{divide-left-mono} \ \text{mult-mono}) \\
\texttt{mult-pos-pos ln-x-over-x-mono power-mono} \ \text{auto} \\
\texttt{also have} \ \ldots \leq -1 + (25 / 36 + 25 / 18) / (2 * 3) + (25 / 18 / 3) / 2 + 1 / (12 * 3^2) \\
\quad \text{using} \ \ln\pi \ \ln\text{-2-25-over-36} \ \ln\text{-3} \ \text{by} \ (\text{intro add-mono} \ \text{divide-left-mono} \ \text{divide-right-mono}) \ \text{auto} \\
\texttt{also have} \ \ldots \leq 0 \ \text{by simp} \\
\texttt{finally have} \ \ln n - \ln (\text{fact } n) / n \geq 0 \ \text{using} \ n \ \text{by} \ (\text{simp add: divide-right-mono}) \\
\texttt{have} \ -\ln (\text{fact } n) \leq -\ln (2 * p \pi n) / 2 - n * \ln n + n \\
\quad \text{using} \ \ln\text{-fact-bounds}(1)[\text{of } n] \ n \ \text{by simp}
also have \( \ln n + \ldots / n = -\ln (2 * \pi) / (2 * n) - (\ln n / n) / 2 + 1 \)
using \( n \) by (simp add: field-simps ln-mult)
also have \( \ldots \leq \theta - 0 + 1 \)
using \( \pi/gl3 n \) by (intro add-mono diff-mono) auto
finally have upper: \( \ln n - \ln (\text{fact } n) / n \leq 1 \)
using \( n \) by (simp add: divide-right-mono)

with \( \ln n - \ln (\text{fact } n) / n \geq 0 \) have fact-bounds: \( \ln n - \ln (\text{fact } n) / n \in \{0..1\} \) by simp

have \( r \leq \text{prime-sum-upto } (\lambda p. \ln p * 1) n \)
using less-imp-le[OF frac-lt-1] unfolding r-def \( \theta \)-def prime-sum-upto-def
by (intro sum-mono mult-left-mono) (auto simp: dest: prime-gt-0-nat)
also have \( \ldots = \theta n \) by (simp add: \( \theta \)-def)
also have \( \ldots < \ln 4 * n \) using \( n \) by (intro \( \theta \)-upper-bound) auto
finally have \( r / n < \ln 4 \) using \( n \) by (simp add: field-simps)
moreover have \( r \geq 0 \) unfolding r-def prime-sum-upto-def
by (intro sum-nonneg mult-nonneg-nonneg) (auto dest: prime-gt-0-nat)
ultimately have r-bounds: \( r / n \in \{0..<\ln 4\} \) by simp

have \( \ln (\text{fact } n :: \text{real}) = \text{sum-upto } (\lambda k. \text{mangoldt } k * \text{real } (n \text{ div } k)) (\text{real } n) \)
by (simp add: ln-fact-conv-sum-upto-mangoldt)
also have \( \ldots = (\sum (p,i) | \text{prime } p \land i > \theta \land \text{real } (p ^ i) \leq \text{real } n.) \)
\( \ln (\text{real } p) * \text{real } (n \text{ div } (p ^ i))\)
by (intro sum-upto-primepow) (auto simp: mangoldt-non-primepow)
also have \( \{(p, i). \text{prime } p \land i > \theta \land \text{real } (p ^ i) \leq \text{real } n \} = \)
\( \{(p, i). \text{prime } p \land i = 1 \land p \leq n \} \cup \)
\( \{(p, i). \text{prime } p \land i > 1 \land (p ^ i) \leq n \} \) unfolding of-nat-le-iff
by (auto simp: not-less le-Suc-eq)
also have \( \sum (p,i)\in\ldots \ln (\text{real } p) * \text{real } (n \text{ div } (p ^ i)) \)
\( = (\sum (p,i) | \text{prime } p \land i = 1 \land p \leq n. \ln (\text{real } p) * \text{real } (n \text{ div } (p ^ i))) \)

+ \( R \)
(\( \text{is } = \:?S + \cdot \))
by (subst sum.union_disjoint) (auto intro!: finite-subset[OF frac-lt-1] simp: R-def)
also have \(?S = \text{prime-sum-upto } (\lambda p. \ln (\text{real } p) * \text{real } (n \text{ div } p)) (\text{real } n) \)
unfolding prime-sum-upto-def
by (intro sum.reindex-bij-witness[of - \( \lambda p. (p, 1) \) fst]) auto
also have \( \ldots = \text{prime-sum-upto } (\lambda p. \ln (\text{real } p) * \text{real } n / \text{real } p) n - r \)
unfolding r-def prime-sum-upto-def sum-subtractf[symmetric] using n
by (intro sum.cong) (auto simp: frac-def real-of-nat-div algebra-simps)
also have \( \text{prime-sum-upto } (\lambda p. \ln (\text{real } p) * \text{real } n / \text{real } p) n = n * \text{M } n \)
by (simp add: primes-M-def sum-distrib-left sum-distrib-right prime-sum-upto-def field-simps)
finally have \( \text{M } n - \ln n = \ln (\text{fact } n) / n - \ln n + r / n - R / n \)
using \( n \) by (simp add: field-simps)
hence \( \ln n - \text{M } n = \ln n - \ln (\text{fact } n) / n - r / n + R / n \)
by simp
with fact-bounds r-bounds R-bounds show \( \text{M } n - \ln n \in \{-1 - 9 / p \pi^2 <..\ln 4\} \)
by simp

109
As a simple corollary, we obtain a similar bound on the reals.

**lemma**  
\text{mertens-bound-real-strong}:  
\begin{align*}  
\text{fixes } x :: \text{real} & \quad \text{assumes } x \geq 1 \\
\text{shows } \forall x. x - \ln x & \in [-\ln 4 - 1 / \pi, 2 - \ln (1 + \frac{x}{\ln (1 + \frac{x}{\ln 4})}) < \ldots \\
\text{proof} & \quad \begin{align*}  
\text{have } \forall x. x - \ln x & \leq \forall x. (\ln (\text{nat } x)) - \ln x \\
\text{using } \text{assms} & \quad \text{by } \text{simp} \\
\text{also have } x & \leq \ln 4 \\
\text{using } \text{mertens-bound-strong}[\text{of } \ln x] & \quad \text{assms } \text{by } \text{simp} \\
\text{finally have } \forall x. x - \ln x & \leq \ln 4 . \\
\end{align*} \\
\text{from } \text{assms} & \quad \text{have } \forall x. \ln x \neq 0 \quad \text{by } \text{linarith} \\
\text{have } \ln x & \leq \ln 4 \\
\text{using } \text{assms } \text{by } (\text{intro divide-nonneg-pos}) & \quad \text{auto} \\
\text{moreover have } \ln x & \leq 1 / 1 \\
\text{using } \text{assms } \text{frac-def}[\text{of } x] & \quad \text{by } (\text{intro frac-def}) \text{ auto} \\
\text{ultimately have } * & \quad \text{frac-def}[\text{of } x] \in \{0..1\} \quad \text{by } \text{auto} \\
\text{have } \ln x & \leq \ln (\text{real } x) \\
\text{using } \text{assms } \text{frac-def}[\text{of } x] & \quad \text{by } (\text{intro frac-def}) \text{ auto} \\
\text{also have } \ln x & \leq \frac{x}{\ln (\text{real } x)} \\
\text{using } \text{assms } \text{frac-def}[\text{of } x] & \quad \text{by } (\text{intro frac-def}) \text{ auto} \\
\text{finally have } \forall x. x - \ln x & \leq \ln 4 \\
\text{using } \text{assms } \text{frac-def}[\text{of } x] & \quad \text{by } \text{simp} \\
\text{with } \forall x. x - \ln x & \leq \ln 4 \quad \text{show } \forall x. \ln (\text{real } x) \quad \text{by } \text{simp} \\
\text{qed} \\
\end{align*} \\
\text{We weaken this estimate a bit to obtain nicer bounds:}  
\begin{align*}  
\text{lemma } \text{mertens-bound-real':} & \quad \begin{align*}  
\text{fixes } x :: \text{real} & \quad \text{assumes } x \geq 1 \\
\text{shows } \forall x. x - \ln x & \in \{-2..25/18\} \\
\text{proof} & \quad \begin{align*}  
\text{have } \forall x. x - \ln x & \leq \ln 4 \\
\text{using } \text{mertens-bound-real}[\text{of } x] & \quad \text{by } \text{simp} \\
\text{also have } x & \leq \frac{25}{18} \\
\text{using } \text{ln-approx-bound}[\text{of } x] & \quad \text{by } \text{simp} \\
\text{finally have } \forall x. x - \ln x & \leq \frac{25}{18} . \\
\end{align*} \\
\text{have } \ln 2 & \quad \ln (2 : \text{real}) \in \{2/3..25/36\} \\
\text{using } \text{ln-approx-bound}[\text{of } x] & \quad \text{by } \text{simp } \text{add: eval-nat-numeral} \\
\text{have } \ln 3 & \quad \ln (3 : \text{real}) \in \{1..10/9\} \\
\text{using } \text{ln-approx-bound}[\text{of } x] & \quad \text{by } \text{simp } \text{add: eval-nat-numeral} \\
\text{have } \ln 5 & \quad \ln (5 : \text{real}) \in \{4/3..76/45\} \\
\text{using } \text{ln-approx-bound}[\text{of } x] & \quad \text{by } \text{simp } \text{add: eval-nat-numeral} \\
\text{have } \ln 7 & \quad \ln (7 : \text{real}) \in \{3/2..15/7\} \\
\text{using } \text{ln-approx-bound}[\text{of } x] & \quad \text{by } \text{simp } \text{add: eval-nat-numeral} \\
\text{have } \ln 11 & \quad \ln (11 : \text{real}) \in \{5/3..290/99\} \\
\end{align*} \\
\end{align*}
Choosing the lower bound -2 is somewhat arbitrary here; it is a trade-off between getting a reasonably tight bound and having to make lots of case distinctions. To get -2 as a lower bound, we have to show the cases up to \( x = 11 \) by case distinction.

- **have** \( M x - \ln x > -2 \)
- **proof (cases \( x \geq 11 \))**
  - **case False**
    - **hence** \( x \in \{1..<2\} \lor x \in \{2..<3\} \lor x \in \{3..<5\} \lor x \in \{5..<7\} \lor x \in \{7..<11\} \)
      - **using \( x \) by force**
      - **thus \( \?thesis \)**
    - **proof (elim disjE)**
      - **assume** \( x : x \in \{1..<2\} \)
      - **hence** \( \ln x - M x \leq \ln 2 - 0 \)
        - **by (intro diff-mono) auto**
      - **also have** \( \ldots < 2 \) **using \( \ln 2 \leq 25 \over 36 \) by simp**
      - **finally show \( \?thesis \) by simp**
    - **next**
      - **assume** \( x : x \in \{2..<3\} \)
      - **hence** \( \lfloor x \rfloor = 2 \) **by (intro floor-unique) auto**
      - **from** \( x \) **have** \( \ln x - M x \leq \ln 3 - \ln 2 / 2 \)
        - **by (intro diff-mono) (auto simp) (eval-M)**
      - **also have** \( \ldots = \ln (9 / 2) / 2 \) **using \( \ln \text{realpow}[of 3 2] \) by simp**
        - **also have** \( \ldots < 2 \) **using \( \ln\text{-approx-bounds}[of 9 / 2 1] \) by (simp add: eval-nat-numeral)**
          - **finally show \( \?thesis \) by simp**
    - **next**
      - **assume** \( x : x \in \{3..<5\} \)
      - **hence** \( M 3 = M x \)
        - **unfolding primes-M-def**
          - **by (intro prime-sum-upto-eqI (where \( a' = 3 \) and \( b' = 4 \)) (auto simp: nat-le-iff le-numeral-iff nat-eq-iff floor-eq-iff))**
          - **also have** \( M 3 = \ln 2 / 2 + \ln 3 / 3 \)
            - **by (simp add: eval-M eval-nat-numeral mark-out-code)**
          - **finally have** \( \lfloor x \rfloor = \ln 2 / 2 + \ln 3 / 3 \ldots \)
            - **from** \( x \) **have** \( \ln x - M x \leq \ln 5 - (\ln 2 / 2 + \ln 3 / 3) \)
              - **by (intro diff-mono) auto**
            - **also have** \( \ldots < 2 \) **using \( \ln 2 \ln 3 \ln 5 \) by simp**
              - **finally show \( \?thesis \) by simp**
      - **next**
        - **assume** \( x : x \in \{5..<7\} \)
          - **hence** \( M 5 = M x \)
            - **unfolding primes-M-def**
              - **by (intro prime-sum-upto-eqI (where \( a' = 5 \) and \( b' = 6 \)) (auto simp: nat-le-iff le-numeral-iff nat-eq-iff floor-eq-iff))**
              - **also have** \( M 5 = \ln 2 / 2 + \ln 3 / 3 + \ln 5 / 5 \)
                - **by (simp add: eval-M eval-nat-numeral mark-out-code)**
finally have \[ \forall x. x = \ln 2 / 2 + \ln 3 / 3 + \ln 5 / 5 \ldots \]

from \( x \) have \( \ln x - \mathcal{M} x \leq \ln 7 - (\ln 2 / 2 + \ln 3 / 3 + \ln 5 / 5) \)
by \((\text{intro diff-mono})\) auto
also have \( \ldots < 2 \) using \( \ln 2 \ln 3 \ln 5 \ln 7 \) by simp
finally show \( \text{thesis} \) by simp

next
assume \( x : x \in \{7..<11\} \)

hence \( \mathcal{M} 7 = \mathcal{M} x \)
unfolding primes-M-def
by \((\text{intro prime-sum-upto-eqI})\)
where \( a' = 7 \) and \( b' = 10 \)
also have \( \mathcal{M} 7 = \ln 2 / 2 + \ln 3 / 3 + \ln 5 / 5 + \ln 7 / 7 \)
by \((\text{simp add: eval-M eval-nat-numeral mark-out-code})\)
finally have \([\text{simp}]: \forall x. x \leq \ln 11 - (\ln 2 / 2 + \ln 3 / 3 + \ln 5 / 5 + \ln 7 / 7) \)
by \((\text{intro diff-mono})\) auto
also have \( \ldots < 2 \) using \( \ln 2 \ln 3 \ln 5 \ln 7 \ln 11 \) by simp
finally show \( \text{thesis} \) by simp

qed

next
case True
have \( \ln x - \mathcal{M} x \leq 1 + 9/\pi^2 + \ln (1 + \frac{x}{\text{real } \lfloor x \rfloor}) \)
using mertens-bound-real-strong[of x] \( x \) by simp
also have \( 1 + \frac{x}{\text{real } \lfloor x \rfloor} \leq 1 + 1 / 11 \)
using True frac-le-1[of x] \( x \) by \((\text{intro add-mono frac-le})\) auto
hence \( \ln (1 + \frac{x}{\text{real } \lfloor x \rfloor}) \leq \ln (1 + 1 / 11) \)
using \( x \) by \((\text{subst ln-le-cancel-iff})\) (auto intro!: add-pos-nonneg)
also have \( \ldots = \ln (12 / 11) \) by simp
also have \( \ldots \leq 1585 / 18216 \)
using ln-approx-bounds[of 12 / 11 1] \( x \) by \((\text{simp add: eval-nat-numeral})\)
also have \( 9 / \pi^2 \leq 9 / 3.141592653588 \leq 2 \)
using pi-approx by \((\text{intro divide-left-mono power-mono mult-pos-pos})\) auto
also have \( 1 + \ldots + 1585 / 18216 < 2 \)
by \((\text{simp add: power2-eq-square})\)
finally show \( \text{thesis} \) by simp

qed

with \( \exists x. x = \ln x \leq 25 / 18 \) show \( \text{thesis} \) by simp

qed

corollary mertens-first-theorem:
fixes \( x :: \text{real} \) assumes \( x : x \geq 1 \)
shows \( |\mathcal{M} x - \ln x| < 2 \)
using mertens-bound-real[of x] \( x \) by \((\text{simp add: abs-if})\)
5.2 Mertens’ Second Theorem

Mertens’ Second Theorem concerns the asymptotics of the Prime Harmonic Series, namely

\[ \sum_{p \leq x} \frac{1}{p} = \ln \ln x + M + O \left( \frac{1}{\ln x} \right) \]

where \( M \approx 0.261497 \) is the Meissel–Mertens constant.

We define the constant in the following way:

**definition meissel-mertens where**

\[ \text{meissel-mertens} = 1 - \ln (\ln 2) + \text{integral} \{2..\} (\lambda t. (\Re t - \ln t) / (t * \ln t - 2)) \]

We will require the value of the integral \( \int_a^\infty \frac{t}{\ln^2 t} \, dt = \frac{1}{\ln a} \) as an upper bound on the remainder term:

**lemma integral-one-over-x-ln-x-squared:**

assumes \( a :: \text{real} > 1 \)

shows set-integrable borel \{a<..\} (\lambda t. 1 / (t * \ln t - 2)) (is \( ?th1 \))

\[ \text{and set-lebesgue-integral borel} \{a<..\} (\lambda t. 1 / (t * \ln t - 2)) = 1 / \ln a \text{ (is \( ?th2 \))} \]

\[ \text{and ((\lambda t. 1 / (t * (\ln t)^2)) has-integral 1 / \ln a)} \{a<..\} \text{ (is \( ?th3 \))} \]

**proof**

have cont: isCont (\( \lambda t. 1 / (t * (\ln t)^2) \)) \( x \text{ if } x > a \text{ for } x \)

using that a by (auto intro!: continuous-intros)

have deriv: ((\( \lambda x. -1 / \ln x \)) has-real-derivative 1 / (x * (\ln x)^2)) (at x) if \( x > a \text{ for } x \)

using that a by (auto intro!: derivative-eq-intros simp: power2-eq-square field-simps)

have lim1: ((\( \lambda x. -1 / \ln x \)) \circ real-of-ereal) \( \longrightarrow -(1 / \ln a) \) (at-right (ereal a))

unfolding ereal-tendsto-simps using a by (real-asympt simp: field-simps)

have lim2: ((\( \lambda x. -1 / \ln x \)) \circ real-of-ereal) \( \longrightarrow 0 \) (at-left \( \infty \))

unfolding ereal-tendsto-simps using a by (real-asympt simp: field-simps)

have set-integrable borel (einterval a \( \infty \)) (\( \lambda t. 1 / (t * \ln t - 2) \))

by (rule interval-integral-FTC-nonneg[OF - deriv cont - lim1 lim2]) (use a in auto)

thus \( ?th1 \) by simp

have interval-lebesgue-integral borel (ereal a) \( \infty \) (\( \lambda t. 1 / (t * \ln t - 2) \)) = 0 - ((1 / \ln a))

by (rule interval-integral-FTC-nonneg[OF - deriv cont - lim1 lim2]) (use a in auto)

thus \( ?th2 \) by (simp add: interval-integral-to-infinity-eq)

have ((\( \lambda t. 1 / (t * \ln t - 2) \)) has-integral

set-lebesgue-integral lebesgue \{a<..\} (\( \lambda t. 1 / (t * \ln t - 2) \)) \{a<..\}

using \( (?th1) \) by (intro has-integral-set-lebesgue)

(auto simp: set-integrable-def integrable-completion)

also have set-lebesgue-integral lebesgue \{a<..\} (\( \lambda t. 1 / (t * \ln t - 2) \)) = 1 / \ln a
We show that the integral in our definition of the Meissel–Mertens constant is well-defined and give an upper bound for its tails:

lemma
assumes a > (1 :: real)
defines r ≡ (λt. (Ξ t - ln t) / (t * ln t ^ 2))
shows integrable-meissel-mertens: set-integrable lborel {a<..} r
  and meissel-mertens-integral-le: norm (integral {a<..} r) ≤ 2 / ln a
proof —
  have *: ((λt. 2 * (1 / (t * ln t ^ 2))) has-integral 2 * (1 / ln a)) {a<..}
    using assms by (intro has-integral-mult-right integral-one-over-x-ln-x-squared)
  auto
  show set-integrable lborel {a<..} r unfolding set-integrable-def
    proof (rule Bochner-Integration.integrable-bound[OF - AE-I2])
      have integrable lborel (λt::real. indicator {a<..} t * (2 * (1 / (t * ln t ^ 2))))
        using integrable-mult-right[of 2, OF integral-one-over-x-ln-x-squared(1)[of a, unfolded set-integrable-def]] assms
        by (simp add: algebra-simps)
      thus integrable lborel (λt::real. indicator {a<..} t *R (2 / (t * ln t ^ 2)))
        by simp
      fix x :: real
      show norm (indicat-real {a<..} x *R r x) ≤
        norm (indicat-real {a<..} x *R (2 / (x * ln x ^ 2)))
        proof (cases x > a)
          case True
          thus ?thesis
            unfolding norm-scaleR norm-mul r-def norm-divide using mertens-first-theorem[of x] assms
            by (intro mult-mono frac-le divide-nonneg-pos) (auto simp: indicator-def)
        qed (auto simp: indicator-def)
        qed (auto simp: r-def)
      hence r integrable-on {a<..}
        by (simp add: set-borel-integral-eq-integral(1))
      hence norm (integral {a<..} r) ≤ integral {a<..} (λx. 2 * (1 / (x * ln x ^ 2)))
        proof (rule integral-norm-bound-integral)
          show (λx. 2 * (1 / (x * (ln x)^2))) integrable-on {a<..}
            using * by (simp add: has-integral-iff)
          fix x assume x ∈ {a<..}
          hence norm (r x) ≤ 2 / (x * (ln x)^2)
            unfolding r-def norm-divide using mertens-first-theorem[of x] assms
            by (intro mult-mono frac-le divide-nonneg-pos) (auto simp: indicator-def)
          thus norm (r x) ≤ 2 * (1 / (x * ln x ^ 2)) by simp
        qed
      thus also have ... = 2 / ln a
using * by (simp add: has-integral-iff)

finally show norm (integral {a<..} r) ≤ 2 / ln a.

qed

lemma integrable-on-meissel-mertens:
  assumes A ⊆ {1..} Inf A > 1 A ∈ sets borel
  shows (λt. (ln t − ln t) / (t * ln t ^ 2)) integrable-on A

proof –
  from assms obtain x where x: 1 < x < Inf A
    using dense by blast
  from assms have bdd-below A by (intro bdd-below[of - 1]) auto
  hence A ⊆ {Inf A..} by (auto simp: cInf-lower)
  also have ... ⊆ {x<..} using x by auto
  finally have*: A ⊆ {x<..}.
  have set-integrable lborel A (λt. (ln t − ln t) / (t * ln t ^ 2))
    by (rule set-integrable-subset[of integrable-meissel-mertens[of x]]) (use x *
      assms in auto)
  thus ?thesis by (simp add: set-borel-integral-eq-integral(1))

qed

lemma meissel-mertens-bounds: |meissel-mertens − 1 + ln (ln 2)| ≤ 2 / ln 2

proof –
  have*: {2..} − {2<..} = {2..real} by auto
  also have negligible ... by simp
  finally have integral {2..} (λt. (ln t − ln t) / (t * (ln t) ^ 2)) =
   integral {2<..} (λt. (ln t − ln t) / (t * (ln t ^ 2))
    by (intro sym[OF integrable-meissel-mertens[of x]]) auto
  also have norm ... ≤ 2 / ln 2
    by (rule meissel-mertens-integral-le) auto
  finally show |meissel-mertens − 1 + ln (ln 2)| ≤ 2 / ln 2
    by (simp add: meissel-mertens-def)

qed

Finally, obtaining Mertens’ second theorem from the first one is nothing but
a routine summation by parts, followed by a use of the above bound:

theorem mertens-second-theorem:
  defines f ≡ prime-sum-upto (λp. 1 / p)
  shows ∀x. x ≥ 2 ⇒ |f x − ln (ln x) − meissel-mertens| ≤ 4 / ln x
  and (λx. f x − ln (ln x) − meissel-mertens) ∈ O(λx. 1 / ln x)

proof –
  define r where r = (λt. (ln t − ln t) / (t * ln t ^ 2))

  { fix x :: real assume x: x > 2
    have ((λt. 1 − 1 / (t * ln t ^ 2))) has-integral R x * (1 / ln x x) − R 2
     (1 / ln 2) −
         (∑n:real − {2<..} ind prime n * (ln n / real n) * (1 / ln n)) {2..x}
    unfolding primes-M-def prime-sum-upto-altdef1 using x
    by (intro partial-summation-strong[of {}])

  115
(auto intro!: continuous-intros derivative-eq-intros simp: power2-eq-square simp flip: has-real-derivative-iff-has-vector-derivative)
also have \( \mathbb{R} x * (1 / \ln x) = \mathbb{R} 2 * (1 / \ln 2) -
(\sum_{n \in \text{real} - 2^{<x}}. \text{ind prime} \ n * (\ln n / n) * (1 / \ln n)) =
\mathbb{R} x / \ln x - (\sum_{n \in \text{insert} 2} (\text{real} - 2^{<x})). \text{ind prime} \ n * (\ln n / n) * (1 / \ln n))
(is - - - ?S)
by (subst sum.insert)
(auto simp: primes-M-def finite-vimage-real-of-nat-greaterThanAtMost eval-prime-sum-upto)
also have \(?S = f x\)
unfolding f-def prime-sum-upto-altdef1 sum-upto-def using x
by (intro sum.mono-neutral-cong-left) (auto simp: not-less numeral-2-eq-2 le-Suc-eq)
finally have \((\lambda t. -\mathbb{R} t / (t * \ln t ^ 2)) \text{ has-integral} (\mathbb{R} x / (\ln x - f x))\) \{2..x\} by simp
from has-integral-neg[OF this]
have \((\lambda t. \mathbb{R} t / (t * \ln t ^ 2)) \text{ has-integral} (f x - \mathbb{R} x / \ln x)\) \{2..x\} by simp
hence \((\lambda t. \mathbb{R} t / (t * \ln t ^ 2) - 1 / (t * \ln t)) \text{ has-integral}
(f x - \mathbb{R} x / \ln x - (\ln (\ln x) - \ln (\ln 2)))\) \{2..x\} using x
by (intro has-integral-diff fundamental-theorem-of-calculus)
(auto simp flip: has-real-derivative-iff-has-vector-derivative intro: derivative-eq-intros)
also have \(?this \leftrightarrow (r \text{ has-integral} (f x - \mathbb{R} x / \ln x - (\ln (\ln x) - \ln (\ln 2))))\) \{2..x\}
by (intro has-integral-cong) (auto simp: r-def field-simps power2-eq-square)
finally have . . .
} note integral = this
define \(R_\mathbb{R}\) where \(R_\mathbb{R} = (\lambda x. \mathbb{R} x - \ln x)\)
have \(\mathbb{R}: \mathbb{R} x = \ln x + R_\mathbb{R} x\) for \(x\) by (simp add: \(R_\mathbb{R}\)-def)
define \(I\) where \(I = (\lambda x. \text{integral} \{x..\} r)\)
define \(C\) where \(C = (1 - \ln (\ln 2) + I 2)\)
have C-altdef: \(C = \text{meissel-mertens}\)
by (simp add: I-def r-def C-def meissel-mertens-def)
show bound: \(|f x - \ln (\ln x) - \text{meissel-mertens}| \leq 4 / \ln x\) if \(x\): \(x \geq 2\) for \(x\)
proof (cases \(x\) = 2)
  case True
  hence \(|f x - \ln (\ln x) - \text{meissel-mertens}| = |I 2 - \ln (\ln 2) - \text{meissel-mertens}|\)
  by (simp add: f-def eval-prime-sum-upto)
  also have \(?\) ≤ 2 / \ln 2 + 1 / 2
  using meissel-mertens-bounds by linarith
  also have \(?\) ≤ 2 / \ln 2 + 2 / \ln 2 using ln2-le-25-over-36
  by (intro add-mono divide-left-mono) auto
finally show \(?thesis using True by simp
next
  case False
  hence $x : x > 2$ using $x$ by simp
  have $\text{integral}\ (\{2, x\}) + I x = \text{integral}\ (\{2, x\} \cup \{x\})\ r$ unfolding $I\text{-def}\ r\text{-def}$ using $x$
    by (intro integral-Un [symmetric] integrable-on-meissel-mertens) (auto simp: max-def r-def)
  also have $\{2, x\} \cup \{x\} = \{2, x\}$ using $x$ by auto
finally have $\epsilon:\ integral\ \{2, x\} + I x = 2 - I x$ unfolding $I\text{-def}$ by simp
have eq: $f x - \ln (\ln x) - C = R_{\mathfrak{M}} x / \ln x - I x$ using $\text{integral}\(OF\ x\)\ x$ by (auto simp: C-def field-simps $\mathfrak{M}$ has-integral-iff $\ast$)
also have $[\ldots] \leq |R_{\mathfrak{M}} x / \ln x| + \text{norm}\ (I x)$ unfolding real-norm-def by (rule abs-triangle-ineq)
also have $|R_{\mathfrak{M}} x / \ln x| \leq 2 / |\ln x|$
  unfolding $R_{\mathfrak{M}}\text{-def}\ abs\text{-divide}$ using mertens-first-theorem[of $x$] $x$
    by (intro divide-right-mono) auto
  also have $\{x\} - \{x<\} = \{x\}$ and $\{x<\} \subseteq \{x\}$ by auto
hence $I x = \text{integral}\ \{x<\}\ r$ unfolding $I\text{-def}$
  by (intro integral-subset-negligible [symmetric]) simp-all
also have $\text{norm}\ \ldots \leq 2 / \ln x$
  using meissel-mertens-integral-le[of $x$] $x$ by (simp add: $r\text{-def}$)
finally show $|f x - \ln (\ln x) - \text{meissel-mertens}| \leq 4 / \ln x$
  using $x$ by (simp add: C-altdef)
qed

have $(\lambda x. f x - \ln (\ln x) - C) \in O(\lambda x. 1 / \ln x)$
proof (intro landau-o.big[of $4$] eventually-mono[OF eventually-ge-at-top[of $2$]])
  fix $x ::\ real\ assume\ x : x \geq 2$
  with bound[OF $x$] show $|f x - \ln (\ln x) - C| \leq 4 * \text{norm}\ (1 / \ln x)$
    by (simp add: C-altdef)
qed (auto intro!: add-pos-nonneg)
thus $(\lambda x. f x - \ln (\ln x) - \text{meissel-mertens}) \in O(\lambda x. 1 / \ln x)$
  by (simp add: C-altdef)
qed

corollary prime-harmonic-asymptotic: prime-sum-upto $(\lambda p. 1 / p) \sim\{\text{at-top}\} (\lambda x. \ln (\ln x))$
proof -
  define $f$ where $f = \text{prime-sum-upto} (\lambda p. 1 / p)$
  have $(\lambda x. f x - \ln (\ln x) - \text{meissel-mertens} + \text{meissel-mertens}) \in o(\lambda x. \ln (\ln x))$
    unfolding $f\text{-def}$
      by (rule sum-in-smallo[OF landau-o.big-small-trans[OF mertens-second-theorem(2)]]) real-asymptotic+
  hence $(\lambda x. f x - \ln (\ln x)) \in o(\lambda x. \ln (\ln x))$
    by simp
  thus $?\text{thesis}\ unfolding\ f\text{-def}$
    by (rule smallo-imp-asymptotic-equivalent)
qed
As a corollary, we get the divergence of the prime harmonic series.

**corollary** prime-harmonic-diverges: filterlim \((\text{prime-\text{sum-upto}} (\lambda p. 1 / p))\) at-top

**using** asymp-equiv-sym\([\text{OF prime-harmonic-asymp-equiv}]\)

**by** (rule asymp-equiv-at-top-transfer) real-asym

end

6 Acknowledgements

Paulson was supported by the ERC Advanced Grant ALEXANDRIA (Project 742178) funded by the European Research Council at the University of Cambridge, UK.

References


