Prime Number Theorem with Remainder Term

Shuhao Song and Bowen Yao

June 20, 2024

Abstract

We have formalized the proof of the Prime Number Theorem with remainder term. This is the first formalized version of PNT with an explicit error term.

There are many useful results in this AFP entry.

First, the main result, prime number theorem with remainder:

$$\pi(x) = \mathrm{Li}(x) + O\left(x \exp\left(-\sqrt{\log x}/3653\right)\right)$$

Second, the zero-free region of the Riemann zeta function:

$$\zeta(\beta + i\gamma) \neq 0 \text{ when } \beta \geq 1 - \frac{1}{952320} \left(\log(|\gamma| + 2)\right)^{-1}$$

Moreover, we proved a revised version of Perron's formula, together with the zero-free region we can prove the main result.

Contents

theory *PNT_Notation*
imports
Prime_Number_Theorem.Prime_Counting_Functions
begin

definition $PNT\_const\_C_1 \equiv 1\ /\ 952320 :: real$

abbreviation *nat_powr*
(**infixr** *nat′_powr 80*)
where
$n\ nat\_powr\ x \equiv (of\_nat\ n)\ powr\ x$

bundle *pnt_notation*
begin
notation $PNT\_const\_C_1$ (C_1)
notation *norm* (‖_‖)
notation *Suc* $(\_{}_+\ [101]\ 100)$
end

bundle *no_pnt_notation*
begin
no_notation $PNT\_const\_C_1$ (C_1)
no_notation *norm* (‖_‖)
no_notation *Suc* $(\_{}_+\ [101]\ 100)$
end

end
theory *PNT_Remainder_Library*
imports
PNT_Notation
begin
unbundle *pnt_notation*

1 Auxiliary library for prime number theorem

1.1 Zeta function

lemma *pre_zeta_1_bound*:
assumes $0 < Re\ s$
shows $\|pre\_zeta\ 1\ s\| \le \|s\|\ /\ Re\ s$
⟨*proof*⟩

lemma *zeta_pole_eq*:
assumes $s \neq 1$
shows $zeta\ s = pre\_zeta\ 1\ s + 1\ /\ (s - 1)$
⟨*proof*⟩

definition *zeta′* **where** $zeta'\ s \equiv pre\_zeta\ 1\ s * (s - 1) + 1$

lemma *zeta′_analytic*:
zeta′ analytic_on UNIV
⟨*proof*⟩

lemma *zeta′_analytic_on* [*analytic_intros*]:

zeta′ analytic_on A ⟨*proof*⟩

lemma *zeta′_holomorphic_on* [*holomorphic_intros*]:
zeta′ holomorphic_on A ⟨*proof*⟩

lemma *zeta_eq_zeta′*:
zeta s = *zeta′ s* / (*s* − *1*)
⟨*proof*⟩

lemma *zeta′_1* [*simp*]: *zeta′ 1* = *1* ⟨*proof*⟩

lemma *zeta_eq_zero_iff_zeta′*:
shows $s \neq 1 \Longrightarrow zeta'\ s = 0 \longleftrightarrow zeta\ s = 0$
⟨*proof*⟩

lemma *zeta′_eq_zero_iff*:
shows $zeta'\ s = 0 \longleftrightarrow zeta\ s = 0 \wedge s \neq 1$
⟨*proof*⟩

lemma *zeta_eq_zero_iff*:
shows $zeta\ s = 0 \longleftrightarrow zeta'\ s = 0 \vee s = 1$
⟨*proof*⟩

1.2 Logarithm derivatives

definition *logderiv f x* ≡ *deriv f x* / *f x*
definition *log_differentiable*
(**infixr** (*log′_differentiable*) *50*)
where
f log_differentiable x ≡ (*f field_differentiable* (*at x*)) ∧ *f x* ≠ *0*

lemma *logderiv_prod′*:
fixes *f* :: *′n* ⇒ *′f* ⇒ *′f* :: *real_normed_field*
assumes *fin*: *finite I*
and *lder*: $\bigwedge i.\ i \in I \Longrightarrow f\ i$ *log_differentiable a*
shows *logderiv* ($\lambda x.\ \prod i{\in}I.\ f\ i\ x$) *a* = ($\sum i{\in}I.$ *logderiv* (*f i*) *a*) (**is** *?P*)
and ($\lambda x.\ \prod i{\in}I.\ f\ i\ x$) *log_differentiable a* (**is** *?Q*)
⟨*proof*⟩

lemma *logderiv_prod*:
fixes *f* :: *′n* ⇒ *′f* ⇒ *′f* :: *real_normed_field*
assumes *lder*: $\bigwedge i.\ i \in I \Longrightarrow f\ i$ *log_differentiable a*
shows *logderiv* ($\lambda x.\ \prod i{\in}I.\ f\ i\ x$) *a* = ($\sum i{\in}I.$ *logderiv* (*f i*) *a*) (**is** *?P*)
and ($\lambda x.\ \prod i{\in}I.\ f\ i\ x$) *log_differentiable a* (**is** *?Q*)
⟨*proof*⟩

lemma *logderiv_mult*:
assumes *f log_differentiable a*
and *g log_differentiable a*
shows *logderiv* ($\lambda z.\ f\ z * g\ z$) *a* = *logderiv f a* + *logderiv g a* (**is** *?P*)
and ($\lambda z.\ f\ z * g\ z$) *log_differentiable a* (**is** *?Q*)
⟨*proof*⟩

lemma *logderiv_cong_ev*:
assumes $\forall_F$ *x in nhds x*. *f x* = *g x*
and *x* = *y*

shows $logderiv\ f\ x = logderiv\ g\ y$
⟨*proof*⟩

lemma *logderiv_linear*:
assumes $z \neq a$
shows $logderiv\ (\lambda w.\ w - a)\ z = 1\ /\ (z - a)$
and $(\lambda w.\ w - z)$ *log_differentiable* a
⟨*proof*⟩

lemma *deriv_shift*:
assumes f *field_differentiable at* $(a + x)$
shows $deriv\ (\lambda t.\ f\ (a + t))\ x = deriv\ f\ (a + x)$
⟨*proof*⟩

lemma *logderiv_shift*:
assumes f *field_differentiable at* $(a + x)$
shows $logderiv\ (\lambda t.\ f\ (a + t))\ x = logderiv\ f\ (a + x)$
⟨*proof*⟩

lemma *logderiv_inverse*:
assumes $x \neq 0$
shows $logderiv\ (\lambda x.\ 1\ /\ x)\ x = -\ 1\ /\ x$
⟨*proof*⟩

lemma *logderiv_zeta_eq_zeta'*:
assumes $s \neq 1$ $zeta\ s \neq 0$
shows $logderiv\ zeta\ s = logderiv\ zeta'\ s - 1\ /\ (s - 1)$
⟨*proof*⟩

lemma *analytic_logderiv* [*analytic_intros*]:
assumes f *analytic_on* A $\bigwedge z.\ z \in A \Longrightarrow f\ z \neq 0$
shows $(\lambda s.\ logderiv\ f\ s)$ *analytic_on* A
⟨*proof*⟩

1.3 Lemmas of integration and integrability

lemma *powr_has_integral*:
fixes $a\ b\ w :: real$
assumes *Hab*: $a \leq b$ **and** *Hw*: $w > 0 \wedge w \neq 1$
shows $((\lambda x.\ w\ powr\ x)$ *has_integral* $w\ powr\ b\ /\ ln\ w - w\ powr\ a\ /\ ln\ w)\ \{a..b\}$
⟨*proof*⟩

lemma *powr_integrable*:
fixes $a\ b\ w :: real$
assumes *Hab*: $a \leq b$ **and** *Hw*: $w > 0 \wedge w \neq 1$
shows $(\lambda x.\ w\ powr\ x)$ *integrable_on* $\{a..b\}$
⟨*proof*⟩

lemma *powr_integral_bound_gt_1*:
fixes $a\ b\ w :: real$
assumes *Hab*: $a \leq b$ **and** *Hw*: $w > 1$
shows $integral\ \{a..b\}\ (\lambda x.\ w\ powr\ x) \leq w\ powr\ b\ /\ |ln\ w|$
⟨*proof*⟩

lemma *powr_integral_bound_lt_1*:
fixes $a\ b\ w :: real$

assumes *Hab*: $a \le b$ **and** *Hw*: $0 < w \wedge w < 1$
shows *integral* $\{a..b\}$ $(\lambda x.\ w\ powr\ x) \le w\ powr\ a\ /\ |ln\ w|$
⟨*proof*⟩

lemma *set_integrableI_bounded*:
fixes $f :: {'a} \Rightarrow {'b}::\{banach,\ second\_countable\_topology\}$
shows $A \in sets\ M$
$\Longrightarrow (\lambda x.\ indicator\ A\ x *_R f\ x) \in borel\_measurable\ M$
$\Longrightarrow emeasure\ M\ A < \infty$
$\Longrightarrow (AE\ x\ in\ M.\ x \in A \longrightarrow norm\ (f\ x) \le B)$
$\Longrightarrow set\_integrable\ M\ A\ f$
⟨*proof*⟩

lemma *integrable_cut′*:
fixes $a\ b\ c :: real$ **and** $f :: real \Rightarrow real$
assumes $a \le b\ b \le c$
and *Hf*: $\bigwedge x.\ a \le x \Longrightarrow f\ integrable\_on\ \{a..x\}$
shows $f\ integrable\_on\ \{b..c\}$
⟨*proof*⟩

lemma *integration_by_part′*:
fixes $a\ b :: real$
and $f\ g :: real \Rightarrow {'a} :: \{real\_normed\_field,\ banach\}$
and $f'\ g' :: real \Rightarrow {'a}$
assumes $a \le b$
and $\bigwedge x.\ x \in \{a..b\} \Longrightarrow (f\ has\_vector\_derivative\ f'\ x)\ (at\ x)$
and $\bigwedge x.\ x \in \{a..b\} \Longrightarrow (g\ has\_vector\_derivative\ g'\ x)\ (at\ x)$
and *int*: $(\lambda x.\ f\ x * g'\ x)\ integrable\_on\ \{a..b\}$
shows $((\lambda x.\ f'\ x * g\ x)\ has\_integral$
$f\ b * g\ b - f\ a * g\ a - integral\{a..b\}\ (\lambda x.\ f\ x * g'\ x))\ \{a..b\}$
⟨*proof*⟩

lemma *integral_bigo*:
fixes $a :: real$ **and** $f\ g :: real \Rightarrow real$
assumes *f_bound*: $f \in O(g)$
and *Hf*: $\bigwedge x.\ a \le x \Longrightarrow f\ integrable\_on\ \{a..x\}$
and *Hf′*: $\bigwedge x.\ a \le x \Longrightarrow (\lambda x.\ |f\ x|)\ integrable\_on\ \{a..x\}$
and *Hg′*: $\bigwedge x.\ a \le x \Longrightarrow (\lambda x.\ |g\ x|)\ integrable\_on\ \{a..x\}$
shows $(\lambda x.\ integral\{a..x\}\ f) \in O(\lambda x.\ 1 + integral\{a..x\}\ (\lambda x.\ |g\ x|))$
⟨*proof*⟩

lemma *integral_linepath_same_Re*:
assumes *Ha*: $Re\ a = Re\ b$
and *Hb*: $Im\ a < Im\ b$
and *Hf*: $(f\ has\_contour\_integral\ x)\ (linepath\ a\ b)$
shows $((\lambda t.\ f\ (Complex\ (Re\ a)\ t) * \mathrm{i})\ has\_integral\ x)\ \{Im\ a..Im\ b\}$
⟨*proof*⟩

1.4 Lemmas on asymptotics

lemma *eventually_at_top_linorderI′*:
fixes $c :: {'a} :: \{no\_top,\ linorder\}$
assumes *h*: $\bigwedge x.\ c < x \Longrightarrow P\ x$
shows *eventually P at_top*
⟨*proof*⟩

lemma *eventually_le_imp_bigo*:
  assumes $\forall_F\ x\ in\ F.\ \|f\ x\| \le g\ x$
  shows $f \in O[F](g)$
$\langle proof \rangle$

lemma *eventually_le_imp_bigo'*:
  assumes $\forall_F\ x\ in\ F.\ \|f\ x\| \le g\ x$
  shows $(\lambda x.\ \|f\ x\|) \in O[F](g)$
$\langle proof \rangle$

lemma *le_imp_bigo*:
  assumes $\bigwedge x.\ \|f\ x\| \le g\ x$
  shows $f \in O[F](g)$
  $\langle proof \rangle$

lemma *le_imp_bigo'*:
  assumes $\bigwedge x.\ \|f\ x\| \le g\ x$
  shows $(\lambda x.\ \|f\ x\|) \in O[F](g)$
  $\langle proof \rangle$

lemma *exp_bigo*:
  fixes $f\ g :: real \Rightarrow real$
  assumes $\forall_F\ x\ in\ at\_top.\ f\ x \le g\ x$
  shows $(\lambda x.\ exp\ (f\ x)) \in O(\lambda x.\ exp\ (g\ x))$
$\langle proof \rangle$

lemma *ev_le_imp_exp_bigo*:
  fixes $f\ g :: real \Rightarrow real$
  assumes *hf*: $\forall_F\ x\ in\ at\_top.\ 0 < f\ x$
    and *hg*: $\forall_F\ x\ in\ at\_top.\ 0 < g\ x$
    and *le*: $\forall_F\ x\ in\ at\_top.\ ln\ (f\ x) \le ln\ (g\ x)$
  shows $f \in O(g)$
$\langle proof \rangle$

lemma *smallo_ln_diverge_1*:
  fixes $f :: real \Rightarrow real$
  assumes *f_ln*: $f \in o(ln)$
  shows $LIM\ x\ at\_top.\ x * exp\ (-\ f\ x) :> at\_top$
$\langle proof \rangle$

lemma *ln_ln_asymp_pos*: $\forall_F\ x :: real\ in\ at\_top.\ 0 < ln\ (ln\ x)$ $\langle proof \rangle$
lemma *ln_asymp_pos*: $\forall_F\ x :: real\ in\ at\_top.\ 0 < ln\ x$ $\langle proof \rangle$
lemma *x_asymp_pos*: $\forall_F\ x :: real\ in\ at\_top.\ 0 < x$ $\langle proof \rangle$

1.5 Lemmas of *floor*, *ceil* and *nat_powr*

lemma *nat_le_self*: $0 \le x \Longrightarrow nat\ (int\ x) \le x$ $\langle proof \rangle$
lemma *floor_le*: $\bigwedge x :: real.\ \lfloor x \rfloor \le x$ $\langle proof \rangle$
lemma *ceil_ge*: $\bigwedge x :: real.\ x \le \lceil x \rceil$ $\langle proof \rangle$

lemma *nat_lt_real_iff*:
  $(n :: nat) < (a :: real) = (n < nat\ \lceil a \rceil)$
$\langle proof \rangle$

lemma *nat_le_real_iff*:
  $(n :: nat) \le (a :: real) = (n < nat\ (\lfloor a \rfloor + 1))$

⟨*proof*⟩

lemma *of_real_nat_power*: *n nat_powr* (*of_real x* :: *complex*) = *of_real* (*n nat_powr x*) **for** *n x*
⟨*proof*⟩

lemma *norm_nat_power*: ‖*n nat_powr* (*s* :: *complex*)‖ = *n powr* (*Re s*)
⟨*proof*⟩

1.6 Elementary estimation of *exp* and *ln*

lemma *ln_when_ge_3*:
1 < *ln x* **if** *3* ≤ *x* **for** *x* :: *real*
⟨*proof*⟩

lemma *exp_lemma_1*:
fixes *x* :: *real*
assumes *1* ≤ *x*
shows *1* + *exp x* ≤ *exp* (*2* ∗ *x*)
⟨*proof*⟩

lemma *ln_bound_1*:
fixes *t* :: *real*
assumes *Ht*: *0* ≤ *t*
shows *ln* (*14* + *4* ∗ *t*) ≤ *4* ∗ *ln* (*t* + *2*)
⟨*proof*⟩

1.7 Miscellaneous lemmas

abbreviation *fds_zeta_complex* :: *complex fds* ≡ *fds_zeta*

lemma *powr_mono_lt_1_cancel*:
fixes *x a b* :: *real*
assumes *Hx*: *0* < *x* ∧ *x* < *1*
shows (*x powr a* ≤ *x powr b*) = (*b* ≤ *a*)
⟨*proof*⟩

abbreviation *mangoldt_real* :: _ ⇒ *real* ≡ *mangoldt*
abbreviation *mangoldt_complex* :: _ ⇒ *complex* ≡ *mangoldt*

lemma *norm_fds_mangoldt_complex*:
⋀*n*. ‖*fds_nth* (*fds mangoldt_complex*) *n*‖ = *mangoldt_real n* ⟨*proof*⟩

lemma *suminf_norm_bound*:
fixes *f* :: *nat* ⇒ *'a* :: *banach*
assumes *summable g*
and ⋀*n*. ‖*f n*‖ ≤ *g n*
shows ‖*suminf f*‖ ≤ (∑*n*. *g n*)
⟨*proof*⟩

lemma C_1*_gt_zero*: *0* < C_1 ⟨*proof*⟩

unbundle *no_pnt_notation*
end
theory *Relation_of_PNTs*
imports
PNT_Remainder_Library

begin
unbundle *pnt_notation*
unbundle *prime_counting_notation*

2 Implication relation of many forms of prime number theorem

definition *rem_est* :: *real* ⇒ *real* ⇒ *real* ⇒ _ **where**
rem_est c m n ≡ $O(\lambda\ x.\ x * exp\ (-c * ln\ x\ powr\ m * ln\ (ln\ x)\ powr\ n))$

definition *Li* :: *real* ⇒ *real* **where** *Li x* ≡ *integral* {*2..x*} (λ*x*. *1* / *ln x*)
definition *PNT_1* **where** *PNT_1 c m n* ≡ ((λ*x*. π *x* − *Li x*) ∈ *rem_est c m n*)
definition *PNT_2* **where** *PNT_2 c m n* ≡ ((λ*x*. ϑ *x* − *x*) ∈ *rem_est c m n*)
definition *PNT_3* **where** *PNT_3 c m n* ≡ ((λ*x*. ψ *x* − *x*) ∈ *rem_est c m n*)

lemma *rem_est_compare_powr*:
  fixes *c m n* :: *real*
  assumes *h*: *0* < *m m* < *1*
  shows (λ*x*. *x powr* (*2* / *3*)) ∈ *rem_est c m n*
  ⟨*proof*⟩

lemma *PNT_3_imp_PNT_2*:
  fixes *c m n* :: *real*
  assumes *h*: *0* < *m m* < *1* **and** *PNT_3 c m n*
  shows *PNT_2 c m n*
⟨*proof*⟩

definition r_1 **where** r_1 *x* ≡ π *x* − *Li x* **for** *x*
definition r_2 **where** r_2 *x* ≡ ϑ *x* − *x* **for** *x*

lemma *pi_represent_by_theta*:
  fixes *x* :: *real*
  assumes *2* ≤ *x*
  shows π *x* = ϑ *x* / (*ln x*) + *integral* {*2..x*} (λ*t*. ϑ *t* / (*t* ∗ $(ln\ t)^2$))
⟨*proof*⟩

lemma *Li_integrate_by_part*:
  fixes *x* :: *real*
  assumes *2* ≤ *x*
  shows
  (λ*x*. *1* / $(ln\ x)^2$) *integrable_on* {*2..x*}
  Li x = *x* / (*ln x*) − *2* / (*ln 2*) + *integral* {*2..x*} (λ*t*. *1* / $(ln\ t)^2$)
⟨*proof*⟩

lemma ϑ*_integrable*:
  fixes *x* :: *real*
  assumes *2* ≤ *x*
  shows (λ*t*. ϑ *t* / (*t* ∗ $(ln\ t)^2$)) *integrable_on* {*2..x*}
⟨*proof*⟩

lemma r_1*_represent_by_*r_2:
  fixes *x* :: *real*
  assumes *Hx*: *2* ≤ *x*
  shows (λ*t*. r_2 *t* / (*t* ∗ $(ln\ t)^2$)) *integrable_on* {*2..x*} (**is** *?P*)

$r_1\ x = r_2\ x\ /\ (ln\ x) + 2\ /\ ln\ 2 + integral\ \{2..x\}\ (\lambda t.\ r_2\ t\ /\ (t * (ln\ t)^2))$ (**is** *?Q*)
⟨*proof*⟩

lemma *exp_integral_asymp*:
fixes $f\ f' :: real \Rightarrow real$
assumes *cf*: $continuous\_on\ \{a..\}\ f$
and *der*: $\bigwedge x.\ a < x \Longrightarrow DERIV\ f\ x :> f'\ x$
and *td*: $((\lambda x.\ x * f'\ x) \longrightarrow 0)\ at\_top$
and *f_ln*: $f \in o(ln)$
shows $(\lambda x.\ integral\ \{a..x\}\ (\lambda t.\ exp\ (-f\ t))) \sim[at\_top]\ (\lambda x.\ x * exp(-f\ x))$
⟨*proof*⟩

lemma *x_mul_exp_larger_than_const*:
fixes $c :: real$ **and** $g :: real \Rightarrow real$
assumes *g_ln*: $g \in o(ln)$
shows $(\lambda x.\ c) \in O(\lambda x.\ x * exp(-g\ x))$
⟨*proof*⟩

lemma *integral_bigo_exp'*:
fixes $a :: real$ **and** $f\ g\ g' :: real \Rightarrow real$
assumes *f_bound*: $f \in O(\lambda x.\ exp(-g\ x))$
and *Hf*: $\bigwedge x.\ a \le x \Longrightarrow f\ integrable\_on\ \{a..x\}$
and *Hf'*: $\bigwedge x.\ a \le x \Longrightarrow (\lambda x.\ |f\ x|)\ integrable\_on\ \{a..x\}$
and *Hg*: $continuous\_on\ \{a..\}\ g$
and *der*: $\bigwedge x.\ a < x \Longrightarrow DERIV\ g\ x :> g'\ x$
and *td*: $((\lambda x.\ x * g'\ x) \longrightarrow 0)\ at\_top$
and *g_ln*: $g \in o(ln)$
shows $(\lambda x.\ integral\{a..x\}\ f) \in O(\lambda x.\ x * exp(-g\ x))$
⟨*proof*⟩

lemma *integral_bigo_exp*:
fixes $a\ b :: real$ **and** $f\ g\ g' :: real \Rightarrow real$
assumes *le*: $a \le b$
and *f_bound*: $f \in O(\lambda x.\ exp(-g\ x))$
and *Hf*: $\bigwedge x.\ a \le x \Longrightarrow f\ integrable\_on\ \{a..x\}$
and *Hf'*: $\bigwedge x.\ b \le x \Longrightarrow (\lambda x.\ |f\ x|)\ integrable\_on\ \{b..x\}$
and *Hg*: $continuous\_on\ \{b..\}\ g$
and *der*: $\bigwedge x.\ b < x \Longrightarrow DERIV\ g\ x :> g'\ x$
and *td*: $((\lambda x.\ x * g'\ x) \longrightarrow 0)\ at\_top$
and *g_ln*: $g \in o(ln)$
shows $(\lambda x.\ integral\ \{a..x\}\ f) \in O(\lambda x.\ x * exp(-g\ x))$
⟨*proof*⟩

lemma *integrate_r₂_estimate*:
fixes $c\ m\ n :: real$
assumes *hm*: $0 < m\ m < 1$
and *h*: $r_2 \in rem\_est\ c\ m\ n$
shows $(\lambda x.\ integral\ \{2..x\}\ (\lambda t.\ r_2\ t\ /\ (t * (ln\ t)^2))) \in rem\_est\ c\ m\ n$
⟨*proof*⟩

lemma r_2*_div_ln_estimate*:
fixes $c\ m\ n :: real$
assumes *hm*: $0 < m\ m < 1$
and *h*: $r_2 \in rem\_est\ c\ m\ n$
shows $(\lambda x.\ r_2\ x\ /\ (ln\ x) + 2\ /\ ln\ 2) \in rem\_est\ c\ m\ n$

⟨*proof*⟩

lemma *PNT_2_imp_PNT_1*:
  fixes *l* :: *real*
  assumes *h*: *0 < m m < 1* **and** *PNT_2 c m n*
  shows *PNT_1 c m n*
⟨*proof*⟩

theorem *PNT_3_imp_PNT_1*:
  fixes *l* :: *real*
  assumes *h* : *0 < m m < 1* **and** *PNT_3 c m n*
  shows *PNT_1 c m n*
  ⟨*proof*⟩

hide_const (**open**) r_1 r_2
unbundle *no_prime_counting_notation*
unbundle *no_pnt_notation*
end
theory *PNT_Complex_Analysis_Lemmas*
imports
  PNT_Remainder_Library
begin
unbundle *pnt_notation*

3 Some basic theorems in complex analysis

3.1 Introduction rules for holomorphic functions and analytic functions

lemma *holomorphic_on_shift* [*holomorphic_intros*]:
  assumes *f holomorphic_on* ((λ*z. s* + *z*) ‘ *A*)
  shows (λ*z. f* (*s* + *z*)) *holomorphic_on A*
⟨*proof*⟩

lemma *holomorphic_logderiv* [*holomorphic_intros*]:
  assumes *f holomorphic_on A open A* ⋀*z. z* ∈ *A* ⟹ *f z* ≠ *0*
  shows (λ*s. logderiv f s*) *holomorphic_on A*
  ⟨*proof*⟩

lemma *holomorphic_glue_to_analytic*:
  assumes *o*: *open S open T*
    and *hf*: *f holomorphic_on S*
    and *hg*: *g holomorphic_on T*
    and *hI*: ⋀*z. z* ∈ *S* ⟹ *z* ∈ *T* ⟹ *f z* = *g z*
    and *hU*: *U* ⊆ *S* ∪ *T*
  obtains *h*
   where *h analytic_on U*
      ⋀*z. z* ∈ *S* ⟹ *h z* = *f z*
      ⋀*z. z* ∈ *T* ⟹ *h z* = *g z*
⟨*proof*⟩

lemma *analytic_on_powr_right* [*analytic_intros*]:
  assumes *f analytic_on s*
  shows (λ*z. w powr f z*) *analytic_on s*
⟨*proof*⟩

3.2 Factorization of analytic function on compact region

definition *not_zero_on* (**infixr** *not′_zero′_on 46*)
  where *f not_zero_on S* ≡ ∃ *z* ∈ *S*. *f z* ≠ *0*

lemma *not_zero_on_obtain*:
  assumes *f not_zero_on S* **and** *S* ⊆ *T*
  obtains *t* **where** *f t* ≠ *0* **and** *t* ∈ *T*
⟨*proof*⟩

lemma *analytic_on_holomorphic_connected*:
  assumes *hf*: *f analytic_on S*
    and *con*: *connected A*
    and *ne*: *ξ* ∈ *A* **and** *AS*: *A* ⊆ *S*
  obtains *T T′* **where**
    f holomorphic_on T f holomorphic_on T′
    open T open T′ A ⊆ *T S* ⊆ *T′ connected T*
⟨*proof*⟩

lemma *analytic_factor_zero*:
  assumes *hf*: *f analytic_on S*
    and *KS*: *K* ⊆ *S* **and** *con*: *connected K*
    and *ξK*: *ξ* ∈ *K* **and** *ξz*: *f ξ* = *0*
    and *nz*: *f not_zero_on K*
  obtains *g r n*
    where *0* < *n 0* < *r*
        g analytic_on S g not_zero_on K
        ⋀*z*. *z* ∈ *S* ⟹ *f z* = (*z* − *ξ*)^*n* ∗ *g z*
        ⋀*z*. *z* ∈ *ball ξ r* ⟹ *g z* ≠ *0*
⟨*proof*⟩

lemma *analytic_compact_finite_zeros*:
  assumes *af*: *f analytic_on S*
    and *KS*: *K* ⊆ *S*
    and *con*: *connected K*
    and *cm*: *compact K*
    and *nz*: *f not_zero_on K*
  shows *finite* {*z* ∈ *K*. *f z* = *0*}
⟨*proof*⟩

3.2.1 Auxiliary propositions for theorem *analytic_factorization*

definition *analytic_factor_p′* **where**
 ‹*analytic_factor_p′ f S K* ≡
  ∃ *g n*. ∃ *α* :: *nat* ⇒ *complex*.
      g analytic_on S
    ∧ (∀ *z* ∈ *K*. *g z* ≠ *0*)
    ∧ (∀ *z* ∈ *S*. *f z* = *g z* ∗ (∏ *k*<*n*. *z* − *α k*))
    ∧ *α* ‘ {..<*n*} ⊆ *K*›

definition *analytic_factor_p* **where**
 ‹*analytic_factor_p F* ≡
  ∀ *f S K*. *f analytic_on S*
   ⟶ *K* ⊆ *S*
   ⟶ *connected K*
   ⟶ *compact K*

$\longrightarrow$ *f not_zero_on K*
$\longrightarrow$ $\{z \in K.\ f\ z = 0\} = F$
$\longrightarrow$ *analytic_factor_p' f S K*›

lemma *analytic_factorization_E*:
shows *analytic_factor_p* {}
⟨*proof*⟩

lemma *analytic_factorization_I*:
assumes *ind*: *analytic_factor_p F*
and *ξni*: $\xi \notin F$
shows *analytic_factor_p* (*insert ξ F*)
⟨*proof*⟩

A nontrivial analytic function on connected compact region can be factorized as a everywhere-non-zero function and linear terms $z - s_0$ for all zeros s_0. Note that the connected assumption of K may be removed, but we remain it just for simplicity of proof.

theorem *analytic_factorization*:
assumes *af*: *f analytic_on S*
and *KS*: $K \subseteq S$
and *con*: *connected K*
and *compact K*
and *f not_zero_on K*
obtains *g n* **and** α :: *nat* ⇒ *complex* **where**
g analytic_on S
$\bigwedge z.\ z \in K \Longrightarrow g\ z \neq 0$
$\bigwedge z.\ z \in S \Longrightarrow f\ z = g\ z * (\prod k<n.\ (z - \alpha\ k))$
α ‘ $\{..<n\} \subseteq K$
⟨*proof*⟩

3.3 Schwarz theorem in complex analysis

lemma *Schwarz_Lemma1*:
fixes *f* :: *complex* ⇒ *complex*
and ξ :: *complex*
assumes *f holomorphic_on ball 0 1*
and $f\ 0 = 0$
and $\bigwedge z.\ \|z\| < 1 \Longrightarrow \|f\ z\| \leq 1$
and $\|\xi\| < 1$
shows $\|f\ \xi\| \leq \|\xi\|$
⟨*proof*⟩

theorem *Schwarz_Lemma2*:
fixes *f* :: *complex* ⇒ *complex*
and ξ :: *complex*
assumes *holf*: *f holomorphic_on ball 0 R*
and *hR*: $0 < R$ **and** *nz*: $f\ 0 = 0$
and *bn*: $\bigwedge z.\ \|z\| < R \Longrightarrow \|f\ z\| \leq 1$
and *ξR*: $\|\xi\| < R$
shows $\|f\ \xi\| \leq \|\xi\|\ /\ R$
⟨*proof*⟩

3.4 Borel-Carathedory theorem

Borel-Carathedory theorem, from book *Theorem 5.5*, *The Theory of Functions*, *E. C. Titchmarsh*

lemma *Borel_Caratheodory1*:
  assumes *hr*: $0 < R$ $0 < r$ $r < R$
    and *f0*: $f\ 0 = 0$
    and *hf*: $\bigwedge z.\ \|z\| < R \Longrightarrow Re\ (f\ z) \le A$
    and *holf*: f *holomorphic_on* (*ball* 0 R)
    and *zr*: $\|z\| \le r$
  shows $\|f\ z\| \le 2*r/(R-r) * A$
⟨*proof*⟩

lemma *Borel_Caratheodory2*:
  assumes *hr*: $0 < R$ $0 < r$ $r < R$
    and *hf*: $\bigwedge z.\ \|z\| < R \Longrightarrow Re\ (f\ z - f\ 0) \le A$
    and *holf*: f *holomorphic_on* (*ball* 0 R)
    and *zr*: $\|z\| \le r$
  shows $\|f\ z - f\ 0\| \le 2*r/(R-r) * A$
⟨*proof*⟩

theorem *Borel_Caratheodory3*:
  assumes *hr*: $0 < R$ $0 < r$ $r < R$
    and *hf*: $\bigwedge w.\ w \in$ *ball* s $R \Longrightarrow Re\ (f\ w - f\ s) \le A$
    and *holf*: f *holomorphic_on* (*ball* s R)
    and *zr*: $z \in$ *ball* s r
  shows $\|f\ z - f\ s\| \le 2*r/(R-r) * A$
⟨*proof*⟩

3.5 Lemma 3.9

These lemmas is referred to the following material: Theorem 3.9, *The Theory of the Riemann Zeta−Function*, *E. C. Titchmarsh*, *D. R. Heath−Brown*.

lemma *lemma_3_9_beta1*:
  fixes f M r s_0
  assumes *zl*: $0 < r$ $0 \le M$
    and *hf*: f *holomorphic_on ball* 0 r
    and *ne*: $\bigwedge z.\ z \in$ *ball* 0 $r \Longrightarrow f\ z \ne 0$
    and *bn*: $\bigwedge z.\ z \in$ *ball* 0 $r \Longrightarrow \|f\ z\ /\ f\ 0\| \le exp\ M$
  shows $\|$*logderiv* $f\ 0\| \le 4 * M\ /\ r$
    and $\forall s \in$ *cball* 0 $(r\ /\ 4)$. $\|$*logderiv* $f\ s\| \le 8 * M\ /\ r$
⟨*proof*⟩

lemma *lemma_3_9_beta1'*:
  fixes f M r s_0
  assumes *zl*: $0 < r$ $0 \le M$
    and *hf*: f *holomorphic_on ball* s r
    and *ne*: $\bigwedge z.\ z \in$ *ball* s $r \Longrightarrow f\ z \ne 0$
    and *bn*: $\bigwedge z.\ z \in$ *ball* s $r \Longrightarrow \|f\ z\ /\ f\ s\| \le exp\ M$
    and *hs*: $z \in$ *cball* s $(r\ /\ 4)$
  shows $\|$*logderiv* $f\ z\| \le 8 * M\ /\ r$
⟨*proof*⟩

lemma *lemma_3_9_beta2*:
  fixes f M r
  assumes *zl*: $0 < r$ $0 \le M$
    and *af*: f *analytic_on cball* 0 r
    and *f0*: $f\ 0 \ne 0$
    and *rz*: $\bigwedge z.\ z \in$ *cball* 0 $r \Longrightarrow Re\ z > 0 \Longrightarrow f\ z \ne 0$

and *bn*: $\bigwedge z.\ z \in cball\ 0\ r \Longrightarrow \|f\ z\ /\ f\ 0\| \leq exp\ M$
and *hg*: $\Gamma \subseteq \{z \in cball\ 0\ (r\ /\ 2).\ f\ z = 0 \wedge Re\ z \leq 0\}$
shows $-\ Re\ (logderiv\ f\ 0) \leq 8 * M\ /\ r + Re\ (\sum z{\in}\Gamma.\ 1\ /\ z)$
⟨*proof*⟩

theorem *lemma_3_9_beta3*:
fixes *f M r* **and** *s* :: *complex*
assumes *zl*: $0 < r\ 0 \leq M$
and *af*: *f analytic_on cball s r*
and *f0*: $f\ s \neq 0$
and *rz*: $\bigwedge z.\ z \in cball\ s\ r \Longrightarrow Re\ z > Re\ s \Longrightarrow f\ z \neq 0$
and *bn*: $\bigwedge z.\ z \in cball\ s\ r \Longrightarrow \|f\ z\ /\ f\ s\| \leq exp\ M$
and *hg*: $\Gamma \subseteq \{z \in cball\ s\ (r\ /\ 2).\ f\ z = 0 \wedge Re\ z \leq Re\ s\}$
shows $-\ Re\ (logderiv\ f\ s) \leq 8 * M\ /\ r + Re\ (\sum z{\in}\Gamma.\ 1\ /\ (z - s))$
⟨*proof*⟩

unbundle *no_pnt_notation*
end
theory *Zeta_Zerofree*
imports
PNT_Complex_Analysis_Lemmas
begin
unbundle *pnt_notation*

4 Zero-free region of zeta function

lemma *cos_inequality_1*:
fixes *x* :: *real*
shows $3 + 4 * cos\ x + cos\ (2 * x) \geq 0$
⟨*proof*⟩

lemma *multiplicative_fds_zeta*:
completely_multiplicative_function (*fds_nth fds_zeta_complex*)
⟨*proof*⟩

lemma *fds_mangoldt_eq*:
fds mangoldt_complex = −(*fds_deriv fds_zeta* / *fds_zeta*)
⟨*proof*⟩

lemma *abs_conv_abscissa_log_deriv*:
abs_conv_abscissa (*fds_deriv fds_zeta_complex* / *fds_zeta*) ≤ *1*
⟨*proof*⟩

lemma *abs_conv_abscissa_mangoldt*:
abs_conv_abscissa (*fds mangoldt_complex*) ≤ *1*
⟨*proof*⟩

lemma
assumes *s*: *Re s* > *1*
shows *eval_fds_mangoldt*: *eval_fds* (*fds mangoldt*) *s* = − *deriv zeta s* / *zeta s*
and *abs_conv_mangoldt*: *fds_abs_converges* (*fds mangoldt*) *s*
⟨*proof*⟩

lemma *sums_mangoldt*:
fixes *s* :: *complex*

assumes *s*: *Re s* > *1*
shows ((λ*n*. *mangoldt n* / *n nat_powr s*) *has_sum* − *deriv zeta s* / *zeta s*) {*1*..}
⟨*proof*⟩

lemma *sums_Re_logderiv_zeta*:
fixes σ *t* :: *real*
assumes *s*: σ > *1*
shows ((λ*n*. *mangoldt_real n* ∗ *n nat_powr* (−σ) ∗ *cos* (*t* ∗ *ln n*))
has_sum Re (− *deriv zeta* (*Complex* σ *t*) / *zeta* (*Complex* σ *t*))) {*1*..}
⟨*proof*⟩

lemma *logderiv_zeta_ineq*:
fixes σ *t* :: *real*
assumes *s*: σ > *1*
shows *3* ∗ *Re* (*logderiv zeta* (*Complex* σ *0*)) + *4* ∗ *Re* (*logderiv zeta* (*Complex* σ *t*))
+ *Re* (*logderiv zeta* (*Complex* σ (*2*∗*t*))) ≤ *0* (**is** *?x* ≤ *0*)
⟨*proof*⟩

lemma *sums_zeta_real*:
fixes *r* :: *real*
assumes *1* < *r*
shows (∑ *n*. (n_+) *powr* −*r*) = *Re* (*zeta r*)
⟨*proof*⟩

lemma *inverse_zeta_bound'*:
assumes *1* < *Re s*
shows ‖*inverse* (*zeta s*)‖ ≤ *Re* (*zeta* (*Re s*))
⟨*proof*⟩

lemma *zeta_bound'*:
assumes *1* < *Re s*
shows ‖*zeta s*‖ ≤ *Re* (*zeta* (*Re s*))
⟨*proof*⟩

lemma *zeta_bound_trivial'*:
assumes *1* / *2* ≤ *Re s* ∧ *Re s* ≤ *2*
and |*Im s*| ≥ *1* / *11*
shows ‖*zeta s*‖ ≤ *12* + *2* ∗ |*Im s*|
⟨*proof*⟩

lemma *zeta_bound_gt_1*:
assumes *1* < *Re s*
shows ‖*zeta s*‖ ≤ *Re s* / (*Re s* − *1*)
⟨*proof*⟩

lemma *zeta_bound_trivial*:
assumes *1* / *2* ≤ *Re s* **and** |*Im s*| ≥ *1* / *11*
shows ‖*zeta s*‖ ≤ *12* + *2* ∗ |*Im s*|
⟨*proof*⟩

lemma *zeta_nonzero_small_imag'*:
assumes |*Im s*| ≤ *13* / *22* **and** *Re s* ≥ *1* / *2* **and** *Re s* < *1*
shows *zeta s* ≠ *0*
⟨*proof*⟩

lemma *zeta_nonzero_small_imag*:
assumes $|Im\ s| \leq 13\ /\ 22$ **and** $Re\ s > 0$ **and** $s \neq 1$
shows $zeta\ s \neq 0$
⟨*proof*⟩

lemma *inverse_zeta_bound*:
assumes $1 < Re\ s$
shows $\|inverse\ (zeta\ s)\| \leq Re\ s\ /\ (Re\ s - 1)$
⟨*proof*⟩

lemma *deriv_zeta_bound*:
fixes s :: *complex*
assumes *Hr*: $0 < r$ **and** *Hs*: $s \neq 1$
and *hB*: $\bigwedge w.\ \|s - w\| = r \Longrightarrow \|pre\_zeta\ 1\ w\| \leq B$
shows $\|deriv\ zeta\ s\| \leq B\ /\ r + 1\ /\ \|s - 1\|^2$
⟨*proof*⟩

lemma *zeta_lower_bound*:
assumes $0 < Re\ s\ s \neq 1$
shows $1\ /\ \|s - 1\| - \|s\|\ /\ Re\ s \leq \|zeta\ s\|$
⟨*proof*⟩

lemma *logderiv_zeta_bound*:
fixes σ :: *real*
assumes $1 < \sigma\ \sigma \leq 23\ /\ 20$
shows $\|logderiv\ zeta\ \sigma\| \leq 5\ /\ 4 * (1\ /\ (\sigma - 1))$
⟨*proof*⟩

lemma *Re_logderiv_zeta_bound*:
fixes σ :: *real*
assumes $1 < \sigma\ \sigma \leq 23\ /\ 20$
shows $Re\ (logderiv\ zeta\ \sigma) \geq -\ 5\ /\ 4 * (1\ /\ (\sigma - 1))$
⟨*proof*⟩

locale *zeta_bound_param* =
fixes $\vartheta\ \varphi$ :: *real* ⇒ *real*
assumes *zeta_bn'*: $\bigwedge z.\ 1 - \vartheta\ (Im\ z) \leq Re\ z \Longrightarrow Im\ z \geq 1\ /\ 11 \Longrightarrow \|zeta\ z\| \leq exp\ (\varphi\ (Im\ z))$
and *ϑ_pos*: $\bigwedge t.\ 0 < \vartheta\ t \wedge \vartheta\ t \leq 1\ /\ 2$
and *φ_pos*: $\bigwedge t.\ 1 \leq \varphi\ t$
and *inv_ϑ*: $\bigwedge t.\ \varphi\ t\ /\ \vartheta\ t \leq 1\ /\ 960 * exp\ (\varphi\ t)$
and *moϑ*: *antimono* ϑ **and** *moφ*: *mono* φ
begin
definition $region \equiv \{z.\ 1 - \vartheta\ (Im\ z) \leq Re\ z \wedge Im\ z \geq 1\ /\ 11\}$
lemma *zeta_bn*: $\bigwedge z.\ z \in region \Longrightarrow \|zeta\ z\| \leq exp\ (\varphi\ (Im\ z))$
⟨*proof*⟩
lemma *ϑ_pos'*: $\bigwedge t.\ 0 < \vartheta\ t \wedge \vartheta\ t \leq 1$
⟨*proof*⟩
lemma *φ_pos'*: $\bigwedge t.\ 0 < \varphi\ t$ ⟨*proof*⟩
end

locale *zeta_bound_param_1* = *zeta_bound_param* +
fixes γ :: *real*
assumes *γ_cnd*: $\gamma \geq 13\ /\ 22$
begin
definition r **where** $r \equiv \vartheta\ (2 * \gamma + 1)$

end

locale *zeta_bound_param_2* = *zeta_bound_param_1* +
  fixes σ δ :: *real*
  assumes σ*_cnd*: $\sigma \geq 1 + exp\ (- \varphi(2 * \gamma + 1))$
    and δ*_cnd*: $\delta = \gamma \vee \delta = 2 * \gamma$
begin
  definition *s* **where** $s \equiv Complex\ \sigma\ \delta$
end

context *zeta_bound_param_2* **begin**
declare *dist_complex_def* [*simp*] *norm_minus_commute* [*simp*]
declare *legacy_Complex_simps* [*simp*]

lemma *cball_lm*:
  assumes $z \in cball\ s\ r$
  shows $r \leq 1$ $|Re\ z - \sigma| \leq r$ $|Im\ z - \delta| \leq r$
    $1\ /\ 11 \leq Im\ z$ $Im\ z \leq 2 * \gamma + r$
⟨*proof*⟩

lemma *cball_in_region*:
  shows $cball\ s\ r \subseteq region$
⟨*proof*⟩

lemma *Re_s_gt_1*:
  shows $1 < Re\ s$
⟨*proof*⟩

lemma *zeta_analytic_on_region*:
  shows *zeta analytic_on region*
  ⟨*proof*⟩

lemma *zeta_div_bound*:
  assumes $z \in cball\ s\ r$
  shows $\|zeta\ z\ /\ zeta\ s\| \leq exp\ (3 * \varphi\ (2 * \gamma + 1))$
⟨*proof*⟩

lemma *logderiv_zeta_bound*:
  shows $Re\ (logderiv\ zeta\ s) \geq - 24 * \varphi\ (2 * \gamma + 1)\ /\ r$
   and $\bigwedge\beta.\ \sigma - r\ /\ 2 \leq \beta \Longrightarrow zeta\ (Complex\ \beta\ \delta) = 0 \Longrightarrow$
    $Re\ (logderiv\ zeta\ s) \geq - 24 * \varphi\ (2 * \gamma + 1)\ /\ r + 1\ /\ (\sigma - \beta)$
⟨*proof*⟩
end

context *zeta_bound_param_1* **begin**
lemma *zeta_nonzero_region'*:
  assumes $1 + 1\ /\ 960 * (r\ /\ \varphi\ (2 * \gamma + 1)) - r\ /\ 2 \leq \beta$
   and $zeta\ (Complex\ \beta\ \gamma) = 0$
  shows $1 - \beta \geq 1\ /\ 29760 * (r\ /\ \varphi\ (2 * \gamma + 1))$
⟨*proof*⟩

lemma *zeta_nonzero_region*:
  assumes $zeta\ (Complex\ \beta\ \gamma) = 0$
  shows $1 - \beta \geq 1\ /\ 29760 * (r\ /\ \varphi\ (2 * \gamma + 1))$
⟨*proof*⟩

end

context *zeta_bound_param* **begin**
theorem *zeta_nonzero_region*:
  assumes *zeta* (*Complex* β γ) = *0* **and** *Complex* β $\gamma \neq 1$
  shows $1 - \beta \geq 1 / 29760 * (\vartheta\ (2 * |\gamma| + 1) / \varphi\ (2 * |\gamma| + 1))$
⟨*proof*⟩
end

lemma *zeta_bound_param_nonneg*:
  fixes ϑ φ :: *real* ⇒ *real*
  assumes *zeta_bn'*: $\bigwedge z.\ 1 - \vartheta\ (Im\ z) \leq Re\ z \Longrightarrow Im\ z \geq 1 / 11 \Longrightarrow \|zeta\ z\| \leq exp\ (\varphi\ (Im\ z))$
    and *ϑ_pos*: $\bigwedge t.\ 0 \leq t \Longrightarrow 0 < \vartheta\ t \wedge \vartheta\ t \leq 1 / 2$
    and *φ_pos*: $\bigwedge t.\ 0 \leq t \Longrightarrow 1 \leq \varphi\ t$
    and *inv_ϑ*: $\bigwedge t.\ 0 \leq t \Longrightarrow \varphi\ t / \vartheta\ t \leq 1 / 960 * exp\ (\varphi\ t)$
    and *moϑ*: $\bigwedge x\ y.\ 0 \leq x \Longrightarrow x \leq y \Longrightarrow \vartheta\ y \leq \vartheta\ x$
    and *moφ*: $\bigwedge x\ y.\ 0 \leq x \Longrightarrow x \leq y \Longrightarrow \varphi\ x \leq \varphi\ y$
  shows *zeta_bound_param* ($\lambda t.\ \vartheta\ (max\ 0\ t)$) ($\lambda t.\ \varphi\ (max\ 0\ t)$)
  ⟨*proof*⟩

interpretation *classical_zeta_bound*:
  zeta_bound_param $\lambda t.\ 1 / 2$ $\lambda t.\ 4 * ln\ (12 + 2 * max\ 0\ t)$
⟨*proof*⟩

theorem *zeta_nonzero_region*:
  assumes *zeta* (*Complex* β γ) = *0* **and** *Complex* β $\gamma \neq 1$
  shows $1 - \beta \geq C_1 / ln\ (|\gamma| + 2)$
⟨*proof*⟩

unbundle *no_pnt_notation*
end
theory *PNT_Subsummable*
imports
  PNT_Remainder_Library
begin
unbundle *pnt_notation*

definition *has_subsum* **where** *has_subsum f S x* ≡ ($\lambda n.$ *if* $n \in S$ *then f n else 0*) *sums x*
definition *subsum* **where** *subsum f S* ≡ $\sum n.$ *if* $n \in S$ *then f n else 0*
definition *subsummable* (**infix** *subsummable 50*)
  where *f subsummable S* ≡ *summable* ($\lambda n.$ *if* $n \in S$ *then f n else 0*)

syntax *_subsum* :: *pttrn* ⇒ *nat set* ⇒ *'a* ⇒ *'a*
  ((2∑ '_ ∈ (_)./ _) [0, 0, 10] 10)
translations
  ∑ ' x∈S. t => CONST subsum (λx. t) S

syntax *_subsum_prop* :: *pttrn* ⇒ *bool* ⇒ *'a* ⇒ *'a*
  ((2∑ '_ | (_)./ _) [0, 0, 10] 10)
translations
  ∑ ' x|P. t => CONST subsum (λx. t) {x. P}

syntax *_subsum_ge* :: *pttrn* ⇒ *nat* ⇒ *'a* ⇒ *'a*
  ((2∑ '_ ≥ _./ _) [0, 0, 10] 10)
translations

$\sum$ ' $x \geq n$. t => *CONST subsum* (λx. t) {n..}

lemma *has_subsum_finite*:
finite F ⟹ *has_subsum f F* (*sum f F*)
⟨*proof*⟩

lemma *has_subsum_If_finite_set*:
assumes *finite F*
shows *has_subsum* (λn. *if* $n \in F$ *then f n else 0*) *A* (*sum f* ($F \cap A$))
⟨*proof*⟩

lemma *has_subsum_If_finite*:
assumes *finite* {$n \in A$. *p n*}
shows *has_subsum* (λn. *if p n then f n else 0*) *A* (*sum f* {$n \in A$. *p n*})
⟨*proof*⟩

lemma *has_subsum_univ*:
f sums v ⟹ *has_subsum f UNIV v*
⟨*proof*⟩

lemma *subsumI*:
fixes *f* :: *nat* ⇒ *'a* :: {*t2_space*, *comm_monoid_add*}
shows *has_subsum f A x* ⟹ *x* = *subsum f A*
⟨*proof*⟩

lemma *has_subsum_summable*:
has_subsum f A x ⟹ *f subsummable A*
⟨*proof*⟩

lemma *subsummable_sums*:
fixes *f* :: *nat* ⇒ *'a* :: {*comm_monoid_add*, *t2_space*}
shows *f subsummable S* ⟹ *has_subsum f S* (*subsum f S*)
⟨*proof*⟩

lemma *has_subsum_diff_finite*:
fixes *S* :: *'a* :: {*topological_ab_group_add*, *t2_space*}
assumes *finite F has_subsum f A S F* ⊆ *A*
shows *has_subsum f* (*A* − *F*) (*S* − *sum f F*)
⟨*proof*⟩

lemma *subsum_split*:
fixes *f* :: *nat* ⇒ *'a* :: {*topological_ab_group_add*, *t2_space*}
assumes *f subsummable A finite F F* ⊆ *A*
shows *subsum f A* = *sum f F* + *subsum f* (*A* − *F*)
⟨*proof*⟩

lemma *has_subsum_zero* [*simp*]: *has_subsum* (λn. *0*) *A 0* ⟨*proof*⟩
lemma *zero_subsummable* [*simp*]: (λn. *0*) *subsummable A* ⟨*proof*⟩
lemma *zero_subsum* [*simp*]: ($\sum$ '$n \in A$. *0* :: *'a* :: {*comm_monoid_add*, *t2_space*}) = *0* ⟨*proof*⟩

lemma *has_subsum_minus*:
fixes *f* :: *nat* ⇒ *'a* :: *real_normed_vector*
assumes *has_subsum f A a has_subsum g A b*
shows *has_subsum* (λn. *f n* − *g n*) *A* (*a* − *b*)
⟨*proof*⟩

lemma *subsum_minus*:
  assumes *f subsummable A g subsummable A*
  shows *subsum f A* − *subsum g A* = (∑'*n*∈*A*. *f n* − *g n* :: '*a* :: *real_normed_vector*)
  ⟨*proof*⟩

lemma *subsummable_minus*:
  assumes *f subsummable A g subsummable A*
  shows (λ*n*. *f n* − *g n* :: '*a* :: *real_normed_vector*) *subsummable A*
  ⟨*proof*⟩

lemma *has_subsum_uminus*:
  assumes *has_subsum f A a*
  shows *has_subsum* (λ*n*. − *f n* :: '*a* :: *real_normed_vector*) *A* (− *a*)
⟨*proof*⟩

lemma *subsum_uminus*:
  f subsummable A ⟹ − *subsum f A* = (∑'*n*∈*A*. − *f n* :: '*a* :: *real_normed_vector*)
  ⟨*proof*⟩

lemma *subsummable_uminus*:
  f subsummable A ⟹ (λ*n*. − *f n* :: '*a* :: *real_normed_vector*) *subsummable A*
  ⟨*proof*⟩

lemma *has_subsum_add*:
  fixes *f* :: *nat* ⇒ '*a* :: *real_normed_vector*
  assumes *has_subsum f A a has_subsum g A b*
  shows *has_subsum* (λ*n*. *f n* + *g n*) *A* (*a* + *b*)
⟨*proof*⟩

lemma *subsum_add*:
  assumes *f subsummable A g subsummable A*
  shows *subsum f A* + *subsum g A* = (∑'*n*∈*A*. *f n* + *g n* :: '*a* :: *real_normed_vector*)
  ⟨*proof*⟩

lemma *subsummable_add*:
  assumes *f subsummable A g subsummable A*
  shows (λ*n*. *f n* + *g n* :: '*a* :: *real_normed_vector*) *subsummable A*
  ⟨*proof*⟩

lemma *subsum_cong*:
  (⋀*x*. *x* ∈ *A* ⟹ *f x* = *g x*) ⟹ *subsum f A* = *subsum g A*
  ⟨*proof*⟩

lemma *subsummable_cong*:
  fixes *f* :: *nat* ⇒ '*a* :: *real_normed_vector*
  shows (⋀*x*. *x* ∈ *A* ⟹ *f x* = *g x*) ⟹ (*f subsummable A*) = (*g subsummable A*)
  ⟨*proof*⟩

lemma *subsum_norm_bound*:
  fixes *f* :: *nat* ⇒ '*a* :: *banach*
  assumes *g subsummable A* ⋀*n*. *n* ∈ *A* ⟹ ‖*f n*‖ ≤ *g n*
  shows ‖*subsum f A*‖ ≤ *subsum g A*
  ⟨*proof*⟩

lemma *eval_fds_subsum*:
  fixes *f* :: *'a* :: {*nat_power*, *banach*, *real_normed_field*} *fds*
  assumes *fds_converges f s*
  shows *has_subsum* (λ*n*. *fds_nth f n* / *nat_power n s*) {*1*..} (*eval_fds f s*)
⟨*proof*⟩

lemma *fds_abs_subsummable*:
  fixes *f* :: *'a* :: {*nat_power*, *banach*, *real_normed_field*} *fds*
  assumes *fds_abs_converges f s*
  shows (λ*n*. ‖*fds_nth f n* / *nat_power n s*‖) *subsummable* {*1*..}
⟨*proof*⟩

lemma *subsum_mult2*:
  fixes *f* :: *nat* ⇒ *'a* :: *real_normed_algebra*
  shows *f subsummable A* ⟹ (∑ ‘*x*∈*A*. *f x* ∗ *c*) = *subsum f A* ∗ *c*
⟨*proof*⟩

lemma *subsummable_mult2*:
  fixes *f* :: *nat* ⇒ *'a* :: *real_normed_algebra*
  assumes *f subsummable A*
  shows (λ*x*. *f x* ∗ *c*) *subsummable A*
⟨*proof*⟩

lemma *subsum_ge_limit*:
  lim (λ*N*. ∑ *n* = *m*..*N*. *f n*) = (∑ ‘*n* ≥ *m*. *f n*)
⟨*proof*⟩

lemma *has_subsum_ge_limit*:
  fixes *f* :: *nat* ⇒ *'a* :: {*t2_space*, *comm_monoid_add*, *topological_space*}
  assumes ((λ*N*. ∑ *n* = *m*..*N*. *f n*) ⟶ *l*) *at_top*
  shows *has_subsum f* {*m*..} *l*
⟨*proof*⟩

lemma *eval_fds_complex*:
  fixes *f* :: *complex fds*
  assumes *fds_converges f s*
  shows *has_subsum* (λ*n*. *fds_nth f n* / *n nat_powr s*) {*1*..} (*eval_fds f s*)
⟨*proof*⟩

lemma *eval_fds_complex_subsum*:
  fixes *f* :: *complex fds*
  assumes *fds_converges f s*
  shows *eval_fds f s* = (∑ ‘*n* ≥ *1*. *fds_nth f n* / *n nat_powr s*)
    (λ*n*. *fds_nth f n* / *n nat_powr s*) *subsummable* {*1*..}
⟨*proof*⟩

lemma *has_sum_imp_has_subsum*:
  fixes *x* :: *'a* :: {*comm_monoid_add*, *t2_space*}
  assumes (*f has_sum x*) *A*
  shows *has_subsum f A x*
⟨*proof*⟩

unbundle *no_pnt_notation*
end
theory *Perron_Formula*

imports
PNT_Remainder_Library
PNT_Subsummable
begin
unbundle *pnt_notation*

5 Perron's formula

This version of Perron's theorem is referenced to: *Perron's Formula and the Prime Number Theorem for Automorphic L−Functions*, *Jianya Liu*, *Y. Ye*

A contour integral estimation lemma that will be used both in proof of Perron's formula and the prime number theorem.

lemma *perron_aux_3′*:
fixes *f* :: *complex* ⇒ *complex* **and** *a b B T* :: *real*
assumes *Ha*: *0* < *a* **and** *Hb*: *0* < *b* **and** *hT*: *0* < *T*
and *Hf*: ⋀*t*. *t* ∈ {−*T*..*T*} ⟹ ‖*f* (*Complex b t*)‖ ≤ *B*
and *Hf′*: (λ*s*. *f s* ∗ *a powr s* / *s*) *contour_integrable_on* (*linepath* (*Complex b* (−*T*)) (*Complex b T*))
shows ‖*1* / (*2* ∗ *pi* ∗ i) ∗ *contour_integral* (*linepath* (*Complex b* (−*T*)) (*Complex b T*)) (λ*s*. *f s* ∗ *a powr s* / *s*)‖
≤ *B* ∗ *a powr b* ∗ *ln* (*1* + *T* / *b*)
⟨*proof*⟩

locale *perron_locale* =
fixes *b B H T x* :: *real* **and** *f* :: *complex fds*
assumes *Hb*: *0* < *b* **and** *hT*: *b* ≤ *T*
and *Hb′*: *abs_conv_abscissa f* < *b*
and *hH*: *2* ≤ *H* **and** *hH′*: *b* + *1* ≤ *H* **and** *Hx*: *0* < *x*
and *hB*: (∑'*n* ≥ *1*. ‖*fds_nth f n*‖ / *n nat_powr b*) ≤ *B*
begin
definition *r* **where** *r a* ≡
if a ≠ *1 then min* (*1* / (*2* ∗ *T* ∗ |*ln a*|)) (*2* + *ln* (*T* / *b*))
else (*2* + *ln* (*T* / *b*))
definition *path* **where** *path* ≡ *linepath* (*Complex b* (−*T*)) (*Complex b T*)
definition *img_path* **where** *img_path* ≡ *path_image path*
definition σ_a **where** σ_a ≡ *abs_conv_abscissa f*
definition *region* **where** *region* = {*n* :: *nat*. *x* − *x* / *H* ≤ *n* ∧ *n* ≤ *x* + *x* / *H*}
definition *F* **where** *F* (*a* :: *real*) ≡
1 / (*2* ∗ *pi* ∗ i) ∗ *contour_integral path* (λ*s*. *a powr s* / *s*) − (*if 1* ≤ *a then 1 else 0*)
definition *F′* **where** *F′* (*n* :: *nat*) ≡ *F* (*x* / *n*)

lemma *hT′*: *0* < *T* ⟨*proof*⟩
lemma *cond*: *0* < *b b* ≤ *T 0* < *T* ⟨*proof*⟩

lemma *perron_integrable*:
assumes (*0* :: *real*) < *a*
shows (λ*s*. *a powr s* / *s*) *contour_integrable_on* (*linepath* (*Complex b* (−*T*)) (*Complex b T*))
⟨*proof*⟩

lemma *perron_aux_1′*:
fixes *U* :: *real*
assumes *hU*: *0* < *U* **and** *Ha*: *1* < *a*
shows ‖*F a*‖ ≤ *1* / *pi* ∗ *a powr b* / (*T* ∗ |*ln a*|) + *a powr* −*U* ∗ *T* / (*pi* ∗ *U*)
⟨*proof*⟩

lemma *perron_aux_1*:
  assumes *Ha*: *1 < a*
  shows ‖*F a*‖ ≤ *1 / pi ∗ a powr b / (T ∗ |ln a|)* (**is** _ ≤ *?x*)
⟨*proof*⟩

lemma *perron_aux_2′*:
  fixes *U* :: *real*
  assumes *hU*: *0 < U b < U* **and** *Ha*: *0 < a ∧ a < 1*
  shows ‖*F a*‖ ≤ *1 / pi ∗ a powr b / (T ∗ |ln a|) + a powr U ∗ T / (pi ∗ U)*
⟨*proof*⟩

lemma *perron_aux_2*:
  assumes *Ha*: *0 < a ∧ a < 1*
  shows ‖*F a*‖ ≤ *1 / pi ∗ a powr b / (T ∗ |ln a|)* (**is** _ ≤ *?x*)
⟨*proof*⟩

lemma *perron_aux_3*:
  assumes *Ha*: *0 < a*
  shows ‖*1 / (2 ∗ pi ∗* i*) ∗ contour_integral path (λs. a powr s / s)*‖ ≤ *a powr b ∗ ln (1 + T / b)*
⟨*proof*⟩

lemma *perron_aux′*:
  assumes *Ha*: *0 < a*
  shows ‖*F a*‖ ≤ *a powr b ∗ r a*
⟨*proof*⟩

lemma *r_bound*:
  assumes *Hn*: *1 ≤ n*
  shows *r (x / n) ≤ H / T + (if n ∈ region then 2 + ln (T / b) else 0)*
⟨*proof*⟩

lemma *perron_aux*:
  assumes *Hn*: *0 < n*
  shows ‖*F′ n*‖ ≤ *1 / n nat_powr b ∗ (x powr b ∗ H / T)*
    + (if n ∈ region then 3 ∗ (2 + ln (T / b)) else 0) (**is** *?P ≤ ?Q*)
⟨*proof*⟩

definition *a* **where** *a n ≡ fds_nth f n*

lemma *finite_region*: *finite region*
  ⟨*proof*⟩

lemma *zero_notin_region*: *0 ∉ region*
  ⟨*proof*⟩

lemma *path_image_conv*:
  assumes *s ∈ img_path*
  shows *conv_abscissa f < s · 1*
⟨*proof*⟩

lemma *converge_on_path*:
  assumes *s ∈ img_path*
  shows *fds_converges f s*
  ⟨*proof*⟩

lemma *summable_on_path*:
  assumes $s \in$ *img_path*
  shows $(\lambda n.\ a\ n\ /\ n\ nat\_powr\ s)$ *subsummable* $\{1..\}$
  ⟨*proof*⟩

lemma *zero_notin_path*:
  shows $0 \notin closed\_segment\ (Complex\ b\ (-\ T))\ (Complex\ b\ T)$
  ⟨*proof*⟩

lemma *perron_bound*:
  $\|\sum `n \geq 1.\ a\ n * F'\ n\| \leq x\ powr\ b * H * B\ /\ T$
    $+\ 3 * (2 + ln\ (T\ /\ b)) * (\sum n \in region.\ \|a\ n\|)$
⟨*proof*⟩

lemma *perron*:
  $(\lambda s.\ eval\_fds\ f\ s * x\ powr\ s\ /\ s)\ contour\_integrable\_on\ path$
  $\|sum\_upto\ a\ x - 1\ /\ (2 * pi * \mathrm{i}) * contour\_integral\ path\ (\lambda s.\ eval\_fds\ f\ s * x\ powr\ s\ /\ s)\|$
    $\leq x\ powr\ b * H * B\ /\ T + 3 * (2 + ln\ (T\ /\ b)) * (\sum n \in region.\ \|a\ n\|)$
⟨*proof*⟩
end

theorem *perron_formula*:
  fixes $b\ B\ H\ T\ x$ :: *real* **and** f :: *complex fds*
  assumes *Hb*: $0 < b$ **and** *hT*: $b \leq T$
    and *Hb'*: $abs\_conv\_abscissa\ f < b$
    and *hH*: $2 \leq H$ **and** *hH'*: $b + 1 \leq H$ **and** *Hx*: $2 \leq x$
    and *hB*: $(\sum `n \geq 1.\ \|fds\_nth\ f\ n\|\ /\ n\ nat\_powr\ b) \leq B$
  shows $(\lambda s.\ eval\_fds\ f\ s * x\ powr\ s\ /\ s)\ contour\_integrable\_on\ (linepath\ (Complex\ b\ (-T))\ (Complex\ b\ T))$
        $\|sum\_upto\ (fds\_nth\ f)\ x - 1\ /\ (2 * pi * \mathrm{i}) *$
          $contour\_integral\ (linepath\ (Complex\ b\ (-T))\ (Complex\ b\ T))\ (\lambda s.\ eval\_fds\ f\ s * x\ powr\ s\ /\ s)\|$
          $\leq x\ powr\ b * H * B\ /\ T + 3 * (2 + ln\ (T\ /\ b)) * (\sum n \mid x - x\ /\ H \leq n \wedge n \leq x + x\ /\ H.\ \|fds\_nth\ f\ n\|)$
⟨*proof*⟩

theorem *perron_asymp*:
  fixes $b\ x$ :: *real*
  assumes *b*: $b > 0\ ereal\ b > abs\_conv\_abscissa\ f$
  assumes *x*: $x \geq 2\ x \notin \mathbb{N}$
  defines $L \equiv (\lambda T.\ linepath\ (Complex\ b\ (-T))\ (Complex\ b\ T))$
  shows $((\lambda T.\ contour\_integral\ (L\ T)\ (\lambda s.\ eval\_fds\ f\ s * of\_real\ x\ powr\ s\ /\ s))$
        $\longrightarrow 2 * pi * \mathrm{i} * sum\_upto\ (\lambda n.\ fds\_nth\ f\ n)\ x)\ at\_top$
⟨*proof*⟩

unbundle *no_pnt_notation*
end
theory *PNT_with_Remainder*
imports
  Relation_of_PNTs
  Zeta_Zerofree
  Perron_Formula
begin
unbundle *pnt_notation*

6 Estimation of the order of $\frac{\zeta'(s)}{\zeta(s)}$

notation *primes_psi* (ψ)

lemma *zeta_div_bound'*:
assumes $1 + exp\ (-\ 4 * ln\ (14 + 4 * t)) \leq \sigma$
and $13\ /\ 22 \leq t$
and $z \in cball\ (Complex\ \sigma\ t)\ (1\ /\ 2)$
shows $\|zeta\ z\ /\ zeta\ (Complex\ \sigma\ t)\| \leq exp\ (12 * ln\ (14 + 4 * t))$
⟨*proof*⟩

lemma *zeta_div_bound*:
assumes $1 + exp\ (-\ 4 * ln\ (14 + 4 * |t|)) \leq \sigma$
and $13\ /\ 22 \leq |t|$
and $z \in cball\ (Complex\ \sigma\ t)\ (1\ /\ 2)$
shows $\|zeta\ z\ /\ zeta\ (Complex\ \sigma\ t)\| \leq exp\ (12 * ln\ (14 + 4 * |t|))$
⟨*proof*⟩

definition C_2 **where** $C_2 \equiv 319979520 :: real$

lemma *C_2_gt_zero*: $0 < C_2$ ⟨*proof*⟩

lemma *logderiv_zeta_order_estimate'*:
$\forall_F\ t\ in\ (abs\ going\_to\ at\_top).$
$\forall \sigma.\ 1 - 1\ /\ 7 * C_1\ /\ ln\ (|t| + 3) \leq \sigma$
$\longrightarrow \|logderiv\ zeta\ (Complex\ \sigma\ t)\| \leq C_2 * (ln\ (|t| + 3))^2$
⟨*proof*⟩

definition C_3 **where**
$C_3 \equiv SOME\ T.\ 0 < T\ \wedge$
$(\forall t.\ T \leq |t| \longrightarrow$
$(\forall \sigma.\ 1 - 1\ /\ 7 * C_1\ /\ ln\ (|t| + 3) \leq \sigma$
$\longrightarrow \|logderiv\ zeta\ (Complex\ \sigma\ t)\| \leq C_2 * (ln\ (|t| + 3))^2))$

lemma *C_3_prop*:
$0 < C_3\ \wedge$
$(\forall t.\ C_3 \leq |t| \longrightarrow$
$(\forall \sigma.\ 1 - 1\ /\ 7 * C_1\ /\ ln\ (|t| + 3) \leq \sigma$
$\longrightarrow \|logderiv\ zeta\ (Complex\ \sigma\ t)\| \leq C_2 * (ln\ (|t| + 3))^2))$
⟨*proof*⟩

lemma *C_3_gt_zero*: $0 < C_3$ ⟨*proof*⟩

lemma *logderiv_zeta_order_estimate*:
assumes $1 - 1\ /\ 7 * C_1\ /\ ln\ (|t| + 3) \leq \sigma\ \ C_3 \leq |t|$
shows $\|logderiv\ zeta\ (Complex\ \sigma\ t)\| \leq C_2 * (ln\ (|t| + 3))^2$
⟨*proof*⟩

definition *zeta_zerofree_region*
where $zeta\_zerofree\_region \equiv \{s.\ s \neq 1 \wedge 1 - C_1\ /\ ln\ (|Im\ s| + 2) < Re\ s\}$
definition *logderiv_zeta_region*
where $logderiv\_zeta\_region \equiv \{s.\ C_3 \leq |Im\ s| \wedge 1 - 1\ /\ 7 * C_1\ /\ ln\ (|Im\ s| + 3) \leq Re\ s\}$
definition *zeta_strip_region*
where $zeta\_strip\_region\ \sigma\ T \equiv \{s.\ s \neq 1 \wedge \sigma \leq Re\ s \wedge |Im\ s| \leq T\}$
definition *zeta_strip_region'*

where $zeta\_strip\_region'\ \sigma\ T \equiv \{s.\ s \neq 1 \wedge \sigma \le Re\ s \wedge C_3 \le |Im\ s| \wedge |Im\ s| \le T\}$

lemma *strip_in_zerofree_region*:
assumes $1 - C_1 / ln\ (T + 2) < \sigma$
shows $zeta\_strip\_region\ \sigma\ T \subseteq zeta\_zerofree\_region$
⟨*proof*⟩

lemma *strip_in_logderiv_zeta_region*:
assumes $1 - 1 / 7 * C_1 / ln\ (T + 3) \le \sigma$
shows $zeta\_strip\_region'\ \sigma\ T \subseteq logderiv\_zeta\_region$
⟨*proof*⟩

lemma *strip_condition_imp*:
assumes $0 \le T\ 1 - 1 / 7 * C_1 / ln\ (T + 3) \le \sigma$
shows $1 - C_1 / ln\ (T + 2) < \sigma$
⟨*proof*⟩

lemma *zeta_zerofree_region*:
assumes $s \in zeta\_zerofree\_region$
shows $zeta\ s \neq 0$
⟨*proof*⟩

lemma *logderiv_zeta_region_estimate*:
assumes $s \in logderiv\_zeta\_region$
shows $\|logderiv\ zeta\ s\| \le C_2 * (ln\ (|Im\ s| + 3))^2$
⟨*proof*⟩

definition C_4 :: *real* **where** $C_4 \equiv 1 / 6666241$

lemma *C_4_prop*:
$\forall_F\ x\ in\ at\_top.\ C_4 / ln\ x \le C_1 / (7 * ln\ (x + 3))$
⟨*proof*⟩

lemma *C_4_gt_zero*: $0 < C_4$ ⟨*proof*⟩

definition *C_5_prop* **where**
$C_5\_prop\ C_5 \equiv$
$0 < C_5 \wedge (\forall_F\ x\ in\ at\_top.\ (\forall t.\ |t| \le x$
$\longrightarrow \|logderiv\ zeta\ (Complex\ (1 - C_4 / ln\ x)\ t)\| \le C_5 * (ln\ x)^2))$

lemma *logderiv_zeta_bound_vertical'*:
$\exists C_5.\ C_5\_prop\ C_5$
⟨*proof*⟩

definition C_5 **where** $C_5 \equiv SOME\ C_5.\ C_5\_prop\ C_5$

lemma
C_5_gt_zero: $0 < C_5$ (**is** *?prop_1*) **and**
logderiv_zeta_bound_vertical:
$\forall_F\ x\ in\ at\_top.\ \forall t.\ |t| \le x$
$\longrightarrow \|logderiv\ zeta\ (Complex\ (1 - C_4 / ln\ x)\ t)\| \le C_5 * (ln\ x)^2$ (**is** *?prop_2*)
⟨*proof*⟩

7 Deducing prime number theorem using Perron's formula

locale *prime_number_theorem* =
  fixes *c ε* :: *real*
  assumes *Hc*: *0 < c* **and** *Hc'*: *c ∗ c < 2 ∗* C_4 **and** *Hε*: *0 < ε 2 ∗ ε < c*
begin
notation *primes_psi* (*ψ*)
definition *H* **where** *H x ≡ exp (c / 2 ∗ (ln x) powr (1 / 2))* **for** *x* :: *real*
definition *T* **where** *T x ≡ exp (c ∗ (ln x) powr (1 / 2))* **for** *x* :: *real*
definition *a* **where** *a x ≡ 1 −* C_4 */ (c ∗ (ln x) powr (1 / 2))* **for** *x* :: *real*
definition *b* **where** *b x ≡ 1 + 1 / (ln x)* **for** *x* :: *real*
definition *B* **where** *B x ≡ 5 / 4 ∗ ln x* **for** *x* :: *real*
definition *f* **where** *f x s ≡ x powr s / s ∗ logderiv zeta s* **for** *x* :: *real* **and** *s* :: *complex*
definition *R* **where** *R x ≡*
  x powr (b x) ∗ H x ∗ B x / T x + 3 ∗ (2 + ln (T x / b x))
  ∗ (∑ n | x − x / H x ≤ n ∧ n ≤ x + x / H x. ‖fds_nth (fds mangoldt_complex) n‖) **for** *x* :: *real*
definition *Rc'* **where** *Rc' ≡ O(λx. x ∗ exp (− (c / 2 − ε) ∗ ln x powr (1 / 2)))*
definition *Rc* **where** *Rc ≡ O(λx. x ∗ exp (− (c / 2 − 2 ∗ ε) ∗ ln x powr (1 / 2)))*
definition z_1 **where** z_1 *x ≡ Complex (a x) (− T x)* **for** *x*
definition z_2 **where** z_2 *x ≡ Complex (b x) (− T x)* **for** *x*
definition z_3 **where** z_3 *x ≡ Complex (b x) (T x)* **for** *x*
definition z_4 **where** z_4 *x ≡ Complex (a x) (T x)* **for** *x*
definition *rect* **where** *rect x ≡ cbox (*z_1 *x) (*z_3 *x)* **for** *x*
definition *rect'* **where** *rect' x ≡ rect x − {1}* **for** *x*
definition P_t **where** P_t *x t ≡ linepath (Complex (a x) t) (Complex (b x) t)* **for** *x t*
definition P_1 **where** P_1 *x ≡ linepath (*z_1 *x) (*z_4 *x)* **for** *x*
definition P_2 **where** P_2 *x ≡ linepath (*z_2 *x) (*z_3 *x)* **for** *x*
definition P_3 **where** P_3 *x ≡* P_t *x (− T x)* **for** *x*
definition P_4 **where** P_4 *x ≡* P_t *x (T x)* **for** *x*
definition P_r **where** P_r *x ≡ rectpath (*z_1 *x) (*z_3 *x)* **for** *x*

lemma *Rc_eq_rem_est*:
  Rc = rem_est (c / 2 − 2 ∗ ε) (1 / 2) 0
⟨*proof*⟩

lemma *residue_f*:
  residue (f x) 1 = − x
⟨*proof*⟩

lemma *rect_in_strip*:
  rect x − {1} ⊆ zeta_strip_region (a x) (T x)
  ⟨*proof*⟩

lemma *rect_in_strip'*:
  {s ∈ rect x. C_3 *≤ |Im s|} ⊆ zeta_strip_region' (a x) (T x)*
  ⟨*proof*⟩

lemma
  rect'_in_zerofree: *∀* $_F$ *x in at_top. rect' x ⊆ zeta_zerofree_region* **and**
  rect_in_logderiv_zeta: *∀* $_F$ *x in at_top. {s ∈ rect x.* C_3 *≤ |Im s|} ⊆ logderiv_zeta_region*
⟨*proof*⟩

lemma *zeta_nonzero_in_rect*:
  ∀ $_F$ *x in at_top. ∀ s. s ∈ rect' x ⟶ zeta s ≠ 0*
  ⟨*proof*⟩

lemma *zero_notin_rect*: $\forall_F$ *x in at_top. 0* $\notin$ *rect′ x*
⟨*proof*⟩

lemma *f_analytic*:
$\forall_F$ *x in at_top. f x analytic_on rect′ x*
⟨*proof*⟩

lemma *path_image_in_rect_1*:
assumes *0* $\leq$ *T x* $\wedge$ *a x* $\leq$ *b x*
shows *path_image* (P_1 *x*) $\subseteq$ *rect x* $\wedge$ *path_image* (P_2 *x*) $\subseteq$ *rect x*
⟨*proof*⟩

lemma *path_image_in_rect_2*:
assumes *0* $\leq$ *T x* $\wedge$ *a x* $\leq$ *b x* $\wedge$ *t* $\in$ {−*T x*..*T x*}
shows *path_image* (P_t *x t*) $\subseteq$ *rect x*
⟨*proof*⟩

definition *path_in_rect′* **where**
path_in_rect′ x ≡
path_image (P_1 *x*) $\subseteq$ *rect′ x* $\wedge$ *path_image* (P_2 *x*) $\subseteq$ *rect′ x* $\wedge$
path_image (P_3 *x*) $\subseteq$ *rect′ x* $\wedge$ *path_image* (P_4 *x*) $\subseteq$ *rect′ x*

lemma *path_image_in_rect′*:
assumes *0* < *T x* $\wedge$ *a x* < *1* $\wedge$ *1* < *b x*
shows *path_in_rect′ x*
⟨*proof*⟩

lemma *asymp_1*:
$\forall_F$ *x in at_top. 0* < *T x* $\wedge$ *a x* < *1* $\wedge$ *1* < *b x*
⟨*proof*⟩

lemma *f_continuous_on*:
$\forall_F$ *x in at_top.* $\forall$ *A*$\subseteq$*rect′ x. continuous_on A* (*f x*)
⟨*proof*⟩

lemma *contour_integrability*:
$\forall_F$ *x in at_top.*
f x contour_integrable_on P_1 *x* $\wedge$ *f x contour_integrable_on* P_2 *x* $\wedge$
f x contour_integrable_on P_3 *x* $\wedge$ *f x contour_integrable_on* P_4 *x*
⟨*proof*⟩

lemma *contour_integral_rectpath′*:
assumes *f x analytic_on* (*rect′ x*) *0* < *T x* $\wedge$ *a x* < *1* $\wedge$ *1* < *b x*
shows *contour_integral* (P_r *x*) (*f x*) = − *2* ∗ *pi* ∗ i ∗ *x*
⟨*proof*⟩

lemma *contour_integral_rectpath*:
$\forall_F$ *x in at_top. contour_integral* (P_r *x*) (*f x*) = − *2* ∗ *pi* ∗ i ∗ *x*
⟨*proof*⟩

lemma *valid_paths*:
valid_path (P_1 *x*) *valid_path* (P_2 *x*) *valid_path* (P_3 *x*) *valid_path* (P_4 *x*)
⟨*proof*⟩

lemma *integral_rectpath_split*:

assumes $f\ x$ *contour_integrable_on* $P_1\ x \wedge f\ x$ *contour_integrable_on* $P_2\ x \wedge$
$f\ x$ *contour_integrable_on* $P_3\ x \wedge f\ x$ *contour_integrable_on* $P_4\ x$
shows *contour_integral* $(P_3\ x)\ (f\ x)$ + *contour_integral* $(P_2\ x)\ (f\ x)$
− *contour_integral* $(P_4\ x)\ (f\ x)$ − *contour_integral* $(P_1\ x)\ (f\ x)$ = *contour_integral* $(P_r\ x)\ (f\ x)$
⟨*proof*⟩

lemma P_2*_eq*:
$\forall_F$ *x in at_top. contour_integral* $(P_2\ x)\ (f\ x) + 2 * pi * \mathrm{i} * x$
= *contour_integral* $(P_1\ x)\ (f\ x)$ − *contour_integral* $(P_3\ x)\ (f\ x)$ + *contour_integral* $(P_4\ x)\ (f\ x)$
⟨*proof*⟩

lemma *estimation_*P_1:
$(\lambda x.\ \|$*contour_integral* $(P_1\ x)\ (f\ x)\|) \in Rc$
⟨*proof*⟩

lemma *estimation_*$P_t{}'$:
assumes *h*:
$1 < x \wedge max\ 1\ C_3 \leq T\ x\ a\ x < 1 \wedge 1 < b\ x$
$\{s \in rect\ x.\ C_3 \leq |Im\ s|\} \subseteq$ *logderiv_zeta_region*
$f\ x$ *contour_integrable_on* $P_3\ x \wedge f\ x$ *contour_integrable_on* $P_4\ x$
and *Ht*: $|t| = T\ x$
shows $\|$*contour_integral* $(P_t\ x\ t)\ (f\ x)\| \leq C_2 * exp\ 1 * x\ /\ T\ x * (ln\ (T\ x + 3))^2 * (b\ x - a\ x)$
⟨*proof*⟩

lemma *estimation_*P_t:
$(\lambda x.\ \|$*contour_integral* $(P_3\ x)\ (f\ x)\|) \in Rc \wedge$
$(\lambda x.\ \|$*contour_integral* $(P_4\ x)\ (f\ x)\|) \in Rc$
⟨*proof*⟩

lemma *Re_path_*P_2:
$\bigwedge z.\ z \in$ *path_image* $(P_2\ x) \Longrightarrow Re\ z = b\ x$
⟨*proof*⟩

lemma *estimation_*P_2:
$(\lambda x.\ \|1\ /\ (2 * pi * \mathrm{i}) *$ *contour_integral* $(P_2\ x)\ (f\ x) + x\|) \in Rc$
⟨*proof*⟩

lemma *estimation_R*:
$R \in Rc$
⟨*proof*⟩

lemma *perron_psi*:
$\forall_F$ *x in at_top.* $\|\psi\ x + 1\ /\ (2 * pi * \mathrm{i}) *$ *contour_integral* $(P_2\ x)\ (f\ x)\| \leq R\ x$
⟨*proof*⟩

lemma *estimation_perron_psi*:
$(\lambda x.\ \|\psi\ x + 1\ /\ (2 * pi * \mathrm{i}) *$ *contour_integral* $(P_2\ x)\ (f\ x)\|) \in Rc$
⟨*proof*⟩

theorem *prime_number_theorem*:
PNT_3 $(c\ /\ 2 - 2 * \varepsilon)\ (1\ /\ 2)\ 0$
⟨*proof*⟩

no_notation *primes_psi* (ψ)
end

unbundle *prime_counting_notation*

theorem *prime_number_theorem*:
  shows $(\lambda x.\ \pi\ x - Li\ x) \in O(\lambda x.\ x * exp\ (-\ 1\ /\ 3653 * (ln\ x)\ powr\ (1\ /\ 2)))$
⟨*proof*⟩

hide_const (**open**) C_3 C_4 C_5
unbundle *no_prime_counting_notation*
unbundle *no_pnt_notation*
end