Ramsey's Theorem

Tom Ridge

March 17, 2025

Abstract

The infinite form of Ramsey's Theorem is proved following Boolos and Jeffrey, Chapter 26.

Contents

1 Ramsey's Theorem

theory *Ramsey*
imports *Main HOL−Library.Infinite-Set HOL−Library.Ramsey*

begin

Please note: this entire development has been updated and incorporated into *HOL−Library.Ramsey* above. Below, some of the results of the original development are linked to their current versions elsewhere in the Isabelle libraries.

1.1 Library lemmas

lemma *infinite-inj-infinite-image*: *infinite* $Z \Longrightarrow$ *inj-on* $f\ Z \Longrightarrow$ *infinite* $(f\ `\ Z)$
using *finite-imageD* **by** *blast*

lemma *infinite-dom-finite-rng*: [| *infinite* A; *finite* $(f\ `\ A)$ |] ==> $\exists\, b \in f\ `\ A.$ *infinite* $\{a : A.\ f\ a = b\}$
by (*simp add*: *pigeonhole-infinite*)

lemma *infinite-mem*: *infinite* $X \Longrightarrow \exists\, x.\ x \in X$
using *finite-insert* **by** *fastforce*

lemma *not-empty-least*: $(Y{::}nat\ set) \neq \{\} \Longrightarrow \exists\, m.\ m \in Y \wedge (\forall\, m'.\ m' \in Y \longrightarrow m \leq m')$
by (*meson Inf-nat-def1 bdd-below-bot cInf-lower*)

1.2 Dependent Choice Variant

lemma *dc*:
assumes *trans*: *trans r*
and *P0*: *P x0*
and *Pstep*: $\bigwedge x.\ P\ x \Longrightarrow \exists\, y.\ P\ y \wedge (x,\ y) \in r$
obtains $f :: nat \Rightarrow {'a}$ **where** $\bigwedge n.\ P\ (f\ n)$ **and** $\bigwedge n\ m.\ n < m \Longrightarrow (f\ n,\ f\ m) \in r$
by (*metis P0 Pstep dependent-choice local.trans*)

1.3 Ramsey's theorem

lemma *ramsey*: $\forall\, (s{::}nat)\ (r{::}nat)\ (YY{::}'a\ set)\ (f{::}'a\ set \Rightarrow nat).$
infinite YY
$\wedge\ (\forall\, X.\ X \subseteq YY \wedge \mathit{finite}\ X \wedge \mathit{card}\ X = r \longrightarrow f\ X < s)$
$\longrightarrow (\exists\, Y'\ t'.$
$Y' \subseteq YY$
$\wedge\ \mathit{infinite}\ Y'$
$\wedge\ t' < s$
$\wedge\ (\forall\, X.\ X \subseteq Y' \wedge \mathit{finite}\ X \wedge \mathit{card}\ X = r \longrightarrow f\ X = t'))$
using *Ramsey* **by** *fastforce*

end