The Hahn-Banach Theorem for Real Vector Spaces

Gertrud Bauer
http://www.in.tum.de/~bauerg/

June 8, 2008

Abstract

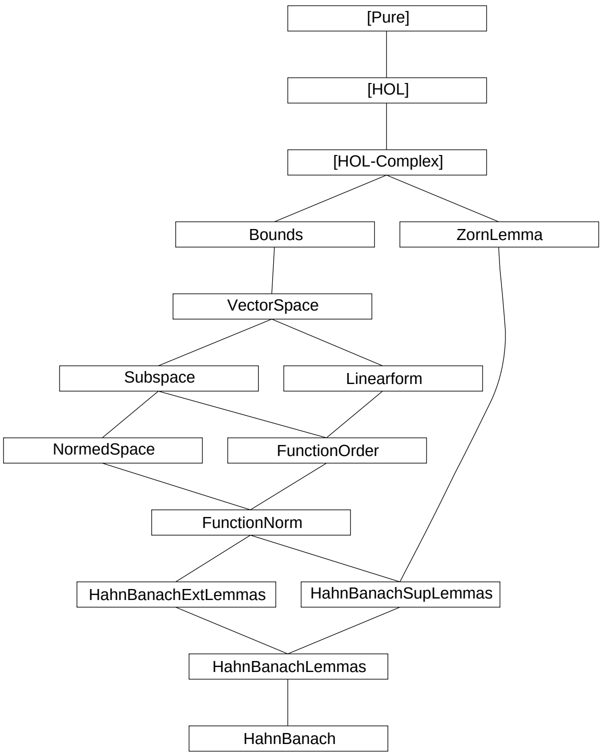
The Hahn-Banach Theorem is one of the most fundamental results in functional analysis. We present a fully formal proof of two versions of the theorem, one for general linear spaces and another for normed spaces. This development is based on simply-typed classical set-theory, as provided by Isabelle/HOL.

Contents

1 Preface

This is a fully formal proof of the Hahn-Banach Theorem. It closely follows the informal presentation given in Heuser's textbook [1, § 36]. Another formal proof of the same theorem has been done in Mizar [3]. A general overview of the relevance and history of the Hahn-Banach Theorem is given by Narici and Beckenstein [2].

The document is structured as follows. The first part contains definitions of basic notions of linear algebra: vector spaces, subspaces, normed spaces, continuous linear-forms, norm of functions and an order on functions by domain extension. The second part contains some lemmas about the supremum (w.r.t. the function order) and extension of non-maximal functions. With these preliminaries, the main proof of the theorem (in its two versions) is conducted in the third part. The dependencies of individual theories are as follows.



Part I

Basic Notions

2 Bounds

theory *Bounds*
imports *Main Real*
begin

locale *lub* =
  fixes *A* **and** *x*
  assumes *least* [*intro?*]: $(\bigwedge a.\ a \in A \Longrightarrow a \leq b) \Longrightarrow x \leq b$
    and *upper* [*intro?*]: $a \in A \Longrightarrow a \leq x$

lemmas [*elim?*] = *lub.least lub.upper*

definition
  the-lub :: *'a::order set* ⇒ *'a* **where**
  the-lub A = *The* (*lub A*)

notation (*xsymbols*)
  the-lub (⨆- [*90*] *90*)

lemma *the-lub-equality* [*elim?*]:
  includes *lub*
  shows ⨆ *A* = (*x*::*'a*::*order*)
⟨*proof*⟩

lemma *the-lubI-ex*:
  assumes *ex*: $\exists x.$ *lub A x*
  shows *lub A* (⨆ *A*)
⟨*proof*⟩

lemma *lub-compat*: *lub A x* = *isLub UNIV A x*
⟨*proof*⟩

lemma *real-complete*:
  fixes *A* :: *real set*
  assumes *nonempty*: $\exists a.\ a \in A$
    and *ex-upper*: $\exists y.\ \forall a \in A.\ a \leq y$
  shows $\exists x.$ *lub A x*
⟨*proof*⟩

end

3 Vector spaces

theory *VectorSpace* **imports** *Real Bounds Zorn* **begin**

3.1 Signature

For the definition of real vector spaces a type $'a$ of the sort $\{plus, minus, zero\}$ is considered, on which a real scalar multiplication $\cdot$ is declared.

consts
prod :: *real* $\Rightarrow$ $'a$::$\{plus, minus, zero\} \Rightarrow 'a$ (**infixr** $'(*')$ *70*)

notation (*xsymbols*)
prod (**infixr** $\cdot$ *70*)
notation (*HTML* **output**)
prod (**infixr** $\cdot$ *70*)

3.2 Vector space laws

A *vector space* is a non-empty set V of elements from $'a$ with the following vector space laws: The set V is closed under addition and scalar multiplication, addition is associative and commutative; $-\ x$ is the inverse of x w. r. t. addition and 0 is the neutral element of addition. Addition and multiplication are distributive; scalar multiplication is associative and the real number 1 is the neutral element of scalar multiplication.

locale *vectorspace* = *var* V +
assumes *non-empty* [*iff*, *intro?*]: $V \neq \{\}$
and *add-closed* [*iff*]: $x \in V \Longrightarrow y \in V \Longrightarrow x + y \in V$
and *mult-closed* [*iff*]: $x \in V \Longrightarrow a \cdot x \in V$
and *add-assoc*: $x \in V \Longrightarrow y \in V \Longrightarrow z \in V \Longrightarrow (x + y) + z = x + (y + z)$
and *add-commute*: $x \in V \Longrightarrow y \in V \Longrightarrow x + y = y + x$
and *diff-self* [*simp*]: $x \in V \Longrightarrow x - x = 0$
and *add-zero-left* [*simp*]: $x \in V \Longrightarrow 0 + x = x$
and *add-mult-distrib1*: $x \in V \Longrightarrow y \in V \Longrightarrow a \cdot (x + y) = a \cdot x + a \cdot y$
and *add-mult-distrib2*: $x \in V \Longrightarrow (a + b) \cdot x = a \cdot x + b \cdot x$
and *mult-assoc*: $x \in V \Longrightarrow (a * b) \cdot x = a \cdot (b \cdot x)$
and *mult-1* [*simp*]: $x \in V \Longrightarrow 1 \cdot x = x$
and *negate-eq1*: $x \in V \Longrightarrow -\ x = (-\ 1) \cdot x$
and *diff-eq1*: $x \in V \Longrightarrow y \in V \Longrightarrow x - y = x + -\ y$

lemma (**in** *vectorspace*) *negate-eq2*: $x \in V \Longrightarrow (-\ 1) \cdot x = -\ x$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *negate-eq2a*: $x \in V \Longrightarrow -1 \cdot x = -\ x$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *diff-eq2*: $x \in V \Longrightarrow y \in V \Longrightarrow x + -\ y = x - y$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *diff-closed* [*iff*]: $x \in V \Longrightarrow y \in V \Longrightarrow x - y \in V$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *neg-closed* [*iff*]: $x \in V \Longrightarrow -\ x \in V$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *add-left-commute*:
$x \in V \Longrightarrow y \in V \Longrightarrow z \in V \Longrightarrow x + (y + z) = y + (x + z)$
$\langle proof \rangle$

theorems (**in** *vectorspace*) *add-ac* =
add-assoc add-commute add-left-commute

The existence of the zero element of a vector space follows from the non-emptiness of carrier set.

lemma (**in** *vectorspace*) *zero* [*iff*]: $0 \in V$
⟨*proof*⟩

lemma (**in** *vectorspace*) *add-zero-right* [*simp*]:
$x \in V \Longrightarrow x + 0 = x$
⟨*proof*⟩

lemma (**in** *vectorspace*) *mult-assoc2*:
$x \in V \Longrightarrow a \cdot b \cdot x = (a * b) \cdot x$
⟨*proof*⟩

lemma (**in** *vectorspace*) *diff-mult-distrib1*:
$x \in V \Longrightarrow y \in V \Longrightarrow a \cdot (x - y) = a \cdot x - a \cdot y$
⟨*proof*⟩

lemma (**in** *vectorspace*) *diff-mult-distrib2*:
$x \in V \Longrightarrow (a - b) \cdot x = a \cdot x - (b \cdot x)$
⟨*proof*⟩

lemmas (**in** *vectorspace*) *distrib* =
add-mult-distrib1 add-mult-distrib2
diff-mult-distrib1 diff-mult-distrib2

Further derived laws:

lemma (**in** *vectorspace*) *mult-zero-left* [*simp*]:
$x \in V \Longrightarrow 0 \cdot x = 0$
⟨*proof*⟩

lemma (**in** *vectorspace*) *mult-zero-right* [*simp*]:
$a \cdot 0 = (0::'a)$
⟨*proof*⟩

lemma (**in** *vectorspace*) *minus-mult-cancel* [*simp*]:
$x \in V \Longrightarrow (- a) \cdot - x = a \cdot x$
⟨*proof*⟩

lemma (**in** *vectorspace*) *add-minus-left-eq-diff*:
$x \in V \Longrightarrow y \in V \Longrightarrow - x + y = y - x$
⟨*proof*⟩

lemma (**in** *vectorspace*) *add-minus* [*simp*]:
$x \in V \Longrightarrow x + - x = 0$
⟨*proof*⟩

lemma (**in** *vectorspace*) *add-minus-left* [*simp*]:
$x \in V \Longrightarrow - x + x = 0$
⟨*proof*⟩

lemma (**in** *vectorspace*) *minus-minus* [*simp*]:
$x \in V \Longrightarrow - (- x) = x$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *minus-zero* [*simp*]:
$- (0::'a) = 0$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *minus-zero-iff* [*simp*]:
$x \in V \Longrightarrow (- x = 0) = (x = 0)$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *add-minus-cancel* [*simp*]:
$x \in V \Longrightarrow y \in V \Longrightarrow x + (- x + y) = y$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *minus-add-cancel* [*simp*]:
$x \in V \Longrightarrow y \in V \Longrightarrow - x + (x + y) = y$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *minus-add-distrib* [*simp*]:
$x \in V \Longrightarrow y \in V \Longrightarrow - (x + y) = - x + - y$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *diff-zero* [*simp*]:
$x \in V \Longrightarrow x - 0 = x$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *diff-zero-right* [*simp*]:
$x \in V \Longrightarrow 0 - x = - x$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *add-left-cancel*:
$x \in V \Longrightarrow y \in V \Longrightarrow z \in V \Longrightarrow (x + y = x + z) = (y = z)$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *add-right-cancel*:
$x \in V \Longrightarrow y \in V \Longrightarrow z \in V \Longrightarrow (y + x = z + x) = (y = z)$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *add-assoc-cong*:
$x \in V \Longrightarrow y \in V \Longrightarrow x' \in V \Longrightarrow y' \in V \Longrightarrow z \in V$
$\Longrightarrow x + y = x' + y' \Longrightarrow x + (y + z) = x' + (y' + z)$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *mult-left-commute*:
$x \in V \Longrightarrow a \cdot b \cdot x = b \cdot a \cdot x$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *mult-zero-uniq*:
$x \in V \Longrightarrow x \neq 0 \Longrightarrow a \cdot x = 0 \Longrightarrow a = 0$
$\langle proof \rangle$

lemma (**in** *vectorspace*) *mult-left-cancel*:
$x \in V \Longrightarrow y \in V \Longrightarrow a \neq 0 \Longrightarrow (a \cdot x = a \cdot y) = (x = y)$
⟨*proof*⟩

lemma (**in** *vectorspace*) *mult-right-cancel*:
$x \in V \Longrightarrow x \neq 0 \Longrightarrow (a \cdot x = b \cdot x) = (a = b)$
⟨*proof*⟩

lemma (**in** *vectorspace*) *eq-diff-eq*:
$x \in V \Longrightarrow y \in V \Longrightarrow z \in V \Longrightarrow (x = z - y) = (x + y = z)$
⟨*proof*⟩

lemma (**in** *vectorspace*) *add-minus-eq-minus*:
$x \in V \Longrightarrow y \in V \Longrightarrow x + y = 0 \Longrightarrow x = - y$
⟨*proof*⟩

lemma (**in** *vectorspace*) *add-minus-eq*:
$x \in V \Longrightarrow y \in V \Longrightarrow x - y = 0 \Longrightarrow x = y$
⟨*proof*⟩

lemma (**in** *vectorspace*) *add-diff-swap*:
$a \in V \Longrightarrow b \in V \Longrightarrow c \in V \Longrightarrow d \in V \Longrightarrow a + b = c + d$
$\Longrightarrow a - c = d - b$
⟨*proof*⟩

lemma (**in** *vectorspace*) *vs-add-cancel-21*:
$x \in V \Longrightarrow y \in V \Longrightarrow z \in V \Longrightarrow u \in V$
$\Longrightarrow (x + (y + z) = y + u) = (x + z = u)$
⟨*proof*⟩

lemma (**in** *vectorspace*) *add-cancel-end*:
$x \in V \Longrightarrow y \in V \Longrightarrow z \in V \Longrightarrow (x + (y + z) = y) = (x = - z)$
⟨*proof*⟩

end

4 Subspaces

theory *Subspace* **imports** *VectorSpace* **begin**

4.1 Definition

A non-empty subset U of a vector space V is a *subspace* of V, iff U is closed under addition and scalar multiplication.

locale *subspace* = *var* U + *var* V +
constrains U :: $'a$::{*minus*, *plus*, *zero*, *uminus*} *set*
assumes *non-empty* [*iff*, *intro*]: $U \neq \{\}$
and *subset* [*iff*]: $U \subseteq V$
and *add-closed* [*iff*]: $x \in U \Longrightarrow y \in U \Longrightarrow x + y \in U$
and *mult-closed* [*iff*]: $x \in U \Longrightarrow a \cdot x \in U$

notation (*symbols*)

subspace (**infix** $\trianglelefteq$ *50*)

declare *vectorspace.intro* [*intro?*] *subspace.intro* [*intro?*]

lemma *subspace-subset* [*elim*]: $U \trianglelefteq V \Longrightarrow U \subseteq V$
⟨*proof*⟩

lemma (**in** *subspace*) *subsetD* [*iff*]: $x \in U \Longrightarrow x \in V$
⟨*proof*⟩

lemma *subspaceD* [*elim*]: $U \trianglelefteq V \Longrightarrow x \in U \Longrightarrow x \in V$
⟨*proof*⟩

lemma *rev-subspaceD* [*elim?*]: $x \in U \Longrightarrow U \trianglelefteq V \Longrightarrow x \in V$
⟨*proof*⟩

lemma (**in** *subspace*) *diff-closed* [*iff*]:
includes *vectorspace*
shows $x \in U \Longrightarrow y \in U \Longrightarrow x - y \in U$
⟨*proof*⟩

Similar as for linear spaces, the existence of the zero element in every subspace follows from the non-emptiness of the carrier set and by vector space laws.

lemma (**in** *subspace*) *zero* [*intro*]:
includes *vectorspace*
shows $0 \in U$
⟨*proof*⟩

lemma (**in** *subspace*) *neg-closed* [*iff*]:
includes *vectorspace*
shows $x \in U \Longrightarrow - x \in U$
⟨*proof*⟩

Further derived laws: every subspace is a vector space.

lemma (**in** *subspace*) *vectorspace* [*iff*]:
includes *vectorspace*
shows *vectorspace* U
⟨*proof*⟩

The subspace relation is reflexive.

lemma (**in** *vectorspace*) *subspace-refl* [*intro*]: $V \trianglelefteq V$
⟨*proof*⟩

The subspace relation is transitive.

lemma (**in** *vectorspace*) *subspace-trans* [*trans*]:
$U \trianglelefteq V \Longrightarrow V \trianglelefteq W \Longrightarrow U \trianglelefteq W$
⟨*proof*⟩

4.2 Linear closure

The *linear closure* of a vector x is the set of all scalar multiples of x.

definition
lin :: $('a$::$\{minus, plus, zero\}) \Rightarrow 'a\ set$ **where**
$lin\ x = \{a \cdot x \mid a.\ True\}$

lemma *linI* [*intro*]: $y = a \cdot x \Longrightarrow y \in lin\ x$
⟨*proof*⟩

lemma *linI'* [*iff*]: $a \cdot x \in lin\ x$
⟨*proof*⟩

lemma *linE* [*elim*]:
$x \in lin\ v \Longrightarrow (\bigwedge a{::}real.\ x = a \cdot v \Longrightarrow C) \Longrightarrow C$
⟨*proof*⟩

Every vector is contained in its linear closure.

lemma (**in** *vectorspace*) *x-lin-x* [*iff*]: $x \in V \Longrightarrow x \in lin\ x$
⟨*proof*⟩

lemma (**in** *vectorspace*) *0-lin-x* [*iff*]: $x \in V \Longrightarrow 0 \in lin\ x$
⟨*proof*⟩

Any linear closure is a subspace.

lemma (**in** *vectorspace*) *lin-subspace* [*intro*]:
$x \in V \Longrightarrow lin\ x \trianglelefteq V$
⟨*proof*⟩

Any linear closure is a vector space.

lemma (**in** *vectorspace*) *lin-vectorspace* [*intro*]:
assumes $x \in V$
shows *vectorspace* $(lin\ x)$
⟨*proof*⟩

4.3 Sum of two vectorspaces

The *sum* of two vectorspaces U and V is the set of all sums of elements from U and V.

instance *fun* :: (*type*, *type*) *plus* ⟨*proof*⟩

defs (**overloaded**)
sum-def: $U + V \equiv \{u + v \mid u\ v.\ u \in U \wedge v \in V\}$

lemma *sumE* [*elim*]:
$x \in U + V \Longrightarrow (\bigwedge u\ v.\ x = u + v \Longrightarrow u \in U \Longrightarrow v \in V \Longrightarrow C) \Longrightarrow C$
⟨*proof*⟩

lemma *sumI* [*intro*]:
$u \in U \Longrightarrow v \in V \Longrightarrow x = u + v \Longrightarrow x \in U + V$
⟨*proof*⟩

lemma *sumI'* [*intro*]:
$u \in U \Longrightarrow v \in V \Longrightarrow u + v \in U + V$
⟨*proof*⟩

U is a subspace of $U + V$.

lemma *subspace-sum1* [*iff*]:
  includes *vectorspace* U + *vectorspace* V
  shows $U \trianglelefteq U + V$
⟨*proof*⟩

The sum of two subspaces is again a subspace.

lemma *sum-subspace* [*intro?*]:
  includes *subspace* U E + *vectorspace* E + *subspace* V E
  shows $U + V \trianglelefteq E$
⟨*proof*⟩

The sum of two subspaces is a vectorspace.

lemma *sum-vs* [*intro?*]:
    $U \trianglelefteq E \Longrightarrow V \trianglelefteq E \Longrightarrow$ *vectorspace* $E \Longrightarrow$ *vectorspace* $(U + V)$
  ⟨*proof*⟩

4.4 Direct sums

The sum of U and V is called *direct*, iff the zero element is the only common element of U and V. For every element x of the direct sum of U and V the decomposition in $x = u + v$ with $u \in U$ and $v \in V$ is unique.

lemma *decomp*:
  includes *vectorspace* E + *subspace* U E + *subspace* V E
  assumes *direct*: $U \cap V = \{0\}$
    and *u1*: $u1 \in U$ **and** *u2*: $u2 \in U$
    and *v1*: $v1 \in V$ **and** *v2*: $v2 \in V$
    and *sum*: $u1 + v1 = u2 + v2$
  shows $u1 = u2 \wedge v1 = v2$
⟨*proof*⟩

An application of the previous lemma will be used in the proof of the Hahn-Banach Theorem (see page **??**): for any element $y + a \cdot x_0$ of the direct sum of a vectorspace H and the linear closure of x_0 the components $y \in H$ and a are uniquely determined.

lemma *decomp-H'*:
  includes *vectorspace* E + *subspace* H E
  assumes *y1*: $y1 \in H$ **and** *y2*: $y2 \in H$
    and *x'*: $x' \notin H$ $x' \in E$ $x' \neq 0$
    and *eq*: $y1 + a1 \cdot x' = y2 + a2 \cdot x'$
  shows $y1 = y2 \wedge a1 = a2$
⟨*proof*⟩

Since for any element $y + a \cdot x'$ of the direct sum of a vectorspace H and the linear closure of x' the components $y \in H$ and a are unique, it follows from $y \in H$ that $a = 0$.

lemma *decomp-H'-H*:
  includes *vectorspace* E + *subspace* H E
  assumes *t*: $t \in H$
    and *x'*: $x' \notin H$ $x' \in E$ $x' \neq 0$
  shows $(SOME\ (y, a).\ t = y + a \cdot x' \wedge y \in H) = (t, 0)$

⟨*proof*⟩

The components $y \in H$ and a in $y + a \cdot x'$ are unique, so the function h' defined by $h'\ (y + a \cdot x') = h\ y + a \cdot \xi$ is definite.

lemma h'-*definite*:
  includes *var H*
  assumes h'-*def*:
    $h' \equiv (\lambda x.\ let\ (y,\ a) = SOME\ (y,\ a).\ (x = y + a \cdot x' \wedge y \in H)$
      $in\ (h\ y) + a * xi)$
    and x: $x = y + a \cdot x'$
  includes *vectorspace E* + *subspace H E*
  assumes y: $y \in H$
    and x': $x' \notin H$ $x' \in E$ $x' \neq 0$
  shows $h'\ x = h\ y + a * xi$
⟨*proof*⟩

end

5 Normed vector spaces

theory *NormedSpace* **imports** *Subspace* **begin**

5.1 Quasinorms

A *seminorm* $\|\cdot\|$ is a function on a real vector space into the reals that has the following properties: it is positive definite, absolute homogenous and subadditive.

locale *norm-syntax* =
  fixes *norm* :: $'a \Rightarrow real$ $(\|\text{-}\|)$

locale *seminorm* = *var V* + *norm-syntax* +
  constrains V :: $'a$::{*minus*, *plus*, *zero*, *uminus*} *set*
  assumes *ge-zero* [*iff?*]: $x \in V \Longrightarrow 0 \leq \|x\|$
    and *abs-homogenous* [*iff?*]: $x \in V \Longrightarrow \|a \cdot x\| = |a| * \|x\|$
    and *subadditive* [*iff?*]: $x \in V \Longrightarrow y \in V \Longrightarrow \|x + y\| \leq \|x\| + \|y\|$

declare *seminorm.intro* [*intro?*]

lemma (**in** *seminorm*) *diff-subadditive*:
  includes *vectorspace*
  shows $x \in V \Longrightarrow y \in V \Longrightarrow \|x - y\| \leq \|x\| + \|y\|$
⟨*proof*⟩

lemma (**in** *seminorm*) *minus*:
  includes *vectorspace*
  shows $x \in V \Longrightarrow \|- x\| = \|x\|$
⟨*proof*⟩

5.2 Norms

A *norm* $\|\cdot\|$ is a seminorm that maps only the *0* vector to *0*.

locale *norm* = *seminorm* +
  assumes *zero-iff* [*iff*]: $x \in V \Longrightarrow (\|x\| = 0) = (x = 0)$

5.3 Normed vector spaces

A vector space together with a norm is called a *normed space.*

locale *normed-vectorspace* = *vectorspace* + *norm*

declare *normed-vectorspace.intro* [*intro?*]

lemma (**in** *normed-vectorspace*) *gt-zero* [*intro?*]:
  $x \in V \Longrightarrow x \neq 0 \Longrightarrow 0 < \|x\|$
⟨*proof*⟩

Any subspace of a normed vector space is again a normed vectorspace.

lemma *subspace-normed-vs* [*intro?*]:
  includes *subspace F E* + *normed-vectorspace E*
  shows *normed-vectorspace F norm*
⟨*proof*⟩

end

6 Linearforms

theory *Linearform* **imports** *VectorSpace* **begin**

A *linear form* is a function on a vector space into the reals that is additive and multiplicative.

locale *linearform* = *var V* + *var f* +
  constrains $V :: {'a}::\{minus,\ plus,\ zero,\ uminus\}\ set$
  assumes *add* [*iff*]: $x \in V \Longrightarrow y \in V \Longrightarrow f\ (x + y) = f\,x + f\,y$
    and *mult* [*iff*]: $x \in V \Longrightarrow f\ (a \cdot x) = a * f\,x$

declare *linearform.intro* [*intro?*]

lemma (**in** *linearform*) *neg* [*iff*]:
  includes *vectorspace*
  shows $x \in V \Longrightarrow f\ (- x) = - f\,x$
⟨*proof*⟩

lemma (**in** *linearform*) *diff* [*iff*]:
  includes *vectorspace*
  shows $x \in V \Longrightarrow y \in V \Longrightarrow f\ (x - y) = f\,x - f\,y$
⟨*proof*⟩

Every linear form yields *0* for the *0* vector.

lemma (**in** *linearform*) *zero* [*iff*]:
  includes *vectorspace*
  shows $f\ 0 = 0$
⟨*proof*⟩

end

7 An order on functions

theory *FunctionOrder* **imports** *Subspace Linearform* **begin**

7.1 The graph of a function

We define the *graph* of a (real) function f with domain F as the set

$$\{(x, f\,x).\ x \in F\}$$

So we are modeling partial functions by specifying the domain and the mapping function. We use the term "function" also for its graph.

types $'a\ graph = ('a \times real)\ set$

definition
$graph :: 'a\ set \Rightarrow ('a \Rightarrow real) \Rightarrow 'a\ graph$ **where**
$graph\ F\ f = \{(x, f\,x) \mid x.\ x \in F\}$

lemma *graphI* [*intro*]: $x \in F \Longrightarrow (x, f\,x) \in graph\ F\ f$
⟨*proof*⟩

lemma *graphI2* [*intro?*]: $x \in F \Longrightarrow \exists\, t \in graph\ F\ f.\ t = (x, f\,x)$
⟨*proof*⟩

lemma *graphE* [*elim?*]:
$(x, y) \in graph\ F\ f \Longrightarrow (x \in F \Longrightarrow y = f\,x \Longrightarrow C) \Longrightarrow C$
⟨*proof*⟩

7.2 Functions ordered by domain extension

A function h' is an extension of h, iff the graph of h is a subset of the graph of h'.

lemma *graph-extI*:
$(\bigwedge x.\ x \in H \Longrightarrow h\ x = h'\ x) \Longrightarrow H \subseteq H'$
$\Longrightarrow graph\ H\ h \subseteq graph\ H'\ h'$
⟨*proof*⟩

lemma *graph-extD1* [*dest?*]:
$graph\ H\ h \subseteq graph\ H'\ h' \Longrightarrow x \in H \Longrightarrow h\ x = h'\ x$
⟨*proof*⟩

lemma *graph-extD2* [*dest?*]:
$graph\ H\ h \subseteq graph\ H'\ h' \Longrightarrow H \subseteq H'$
⟨*proof*⟩

7.3 Domain and function of a graph

The inverse functions to *graph* are *domain* and *funct*.

definition
$domain :: {'a}\ graph \Rightarrow {'a}\ set$ **where**
$domain\ g = \{x.\ \exists\, y.\ (x,\ y) \in g\}$

definition
$funct :: {'a}\ graph \Rightarrow ({'a} \Rightarrow real)$ **where**
$funct\ g = (\lambda x.\ (SOME\ y.\ (x,\ y) \in g))$

The following lemma states that g is the graph of a function if the relation induced by g is unique.

lemma *graph-domain-funct*:
assumes $uniq$: $\bigwedge x\ y\ z.\ (x,\ y) \in g \Longrightarrow (x,\ z) \in g \Longrightarrow z = y$
shows $graph\ (domain\ g)\ (funct\ g) = g$
$\langle proof \rangle$

7.4 Norm-preserving extensions of a function

Given a linear form f on the space F and a seminorm p on E. The set of all linear extensions of f, to superspaces H of F, which are bounded by p, is defined as follows.

definition
$norm\text{-}pres\text{-}extensions ::$
${'a}::\{plus,\ minus,\ uminus,\ zero\}\ set \Rightarrow ({'a} \Rightarrow real) \Rightarrow {'a}\ set \Rightarrow ({'a} \Rightarrow real)$
$\Rightarrow {'a}\ graph\ set$ **where**
$norm\text{-}pres\text{-}extensions\ E\ p\ F\ f$
$= \{g.\ \exists\, H\ h.\ g = graph\ H\ h$
$\wedge\ linearform\ H\ h$
$\wedge\ H \trianglelefteq E$
$\wedge\ F \trianglelefteq H$
$\wedge\ graph\ F\ f \subseteq graph\ H\ h$
$\wedge\ (\forall\, x \in H.\ h\ x \leq p\ x)\}$

lemma *norm-pres-extensionE* [*elim*]:
$g \in norm\text{-}pres\text{-}extensions\ E\ p\ F\ f$
$\Longrightarrow (\bigwedge H\ h.\ g = graph\ H\ h \Longrightarrow linearform\ H\ h$
$\Longrightarrow H \trianglelefteq E \Longrightarrow F \trianglelefteq H \Longrightarrow graph\ F\ f \subseteq graph\ H\ h$
$\Longrightarrow \forall\, x \in H.\ h\ x \leq p\ x \Longrightarrow C) \Longrightarrow C$
$\langle proof \rangle$

lemma *norm-pres-extensionI2* [*intro*]:
$linearform\ H\ h \Longrightarrow H \trianglelefteq E \Longrightarrow F \trianglelefteq H$
$\Longrightarrow graph\ F\ f \subseteq graph\ H\ h \Longrightarrow \forall\, x \in H.\ h\ x \leq p\ x$
$\Longrightarrow graph\ H\ h \in norm\text{-}pres\text{-}extensions\ E\ p\ F\ f$
$\langle proof \rangle$

lemma *norm-pres-extensionI*:
$\exists\, H\ h.\ g = graph\ H\ h$
$\wedge\ linearform\ H\ h$
$\wedge\ H \trianglelefteq E$
$\wedge\ F \trianglelefteq H$
$\wedge\ graph\ F\ f \subseteq graph\ H\ h$
$\wedge\ (\forall\, x \in H.\ h\ x \leq p\ x) \Longrightarrow g \in norm\text{-}pres\text{-}extensions\ E\ p\ F\ f$
$\langle proof \rangle$

end

8 The norm of a function

theory *FunctionNorm* **imports** *NormedSpace FunctionOrder* **begin**

8.1 Continuous linear forms

A linear form f on a normed vector space $(V, \|\cdot\|)$ is *continuous*, iff it is bounded, i.e.

$$\exists\, c \in R.\ \forall\, x \in V.\ |f\ x| \le c \cdot \|x\|$$

In our application no other functions than linear forms are considered, so we can define continuous linear forms as bounded linear forms:

locale *continuous* = *var* V + *norm-syntax* + *linearform* +
  assumes *bounded*: $\exists\, c.\ \forall\, x \in V.\ |f\ x| \le c * \|x\|$

declare *continuous.intro* [*intro?*] *continuous-axioms.intro* [*intro?*]

lemma *continuousI* [*intro*]:
  includes *norm-syntax* + *linearform*
  assumes r: $\bigwedge x.\ x \in V \Longrightarrow |f\ x| \le c * \|x\|$
  shows *continuous V norm f*
⟨*proof*⟩

8.2 The norm of a linear form

The least real number c for which holds

$$\forall\, x \in V.\ |f\ x| \le c \cdot \|x\|$$

is called the *norm* of *f*.
For non-trivial vector spaces $V \ne \{0\}$ the norm can be defined as

$$\|f\| = \sup x \ne 0.\ |f\ x|\ /\ \|x\|$$

For the case $V = \{0\}$ the supremum would be taken from an empty set. Since $\mathbb{R}$ is unbounded, there would be no supremum. To avoid this situation it must be guaranteed that there is an element in this set. This element must be $\{\} \ge 0$ so that *fn-norm* has the norm properties. Furthermore it does not have to change the norm in all other cases, so it must be 0, as all other elements are $\{\} \ge 0$.
Thus we define the set B where the supremum is taken from as follows:

$$\{0\} \cup \{|f\ x|\ /\ \|x\|.\ x \ne 0 \wedge x \in F\}$$

fn-norm is equal to the supremum of B, if the supremum exists (otherwise it is undefined).

locale *fn-norm* = *norm-syntax* +
  fixes B **defines** $B\ V f \equiv \{0\} \cup \{|f\,x| \,/\, \|x\| \mid x.\ x \neq 0 \wedge x \in V\}$
  fixes *fn-norm* ($\|$-$\|$-- [0, 1000] 999)
  defines $\|f\|$-$V \equiv \bigsqcup(B\ V f)$

lemma (**in** *fn-norm*) *B-not-empty* [*intro*]: $0 \in B\ V f$
  ⟨*proof*⟩

The following lemma states that every continuous linear form on a normed space $(V, \|\cdot\|)$ has a function norm.

lemma (**in** *normed-vectorspace*) *fn-norm-works*:
  includes *fn-norm* + *continuous*
  shows *lub* $(B\ V f)$ $(\|f\|$-$V)$
⟨*proof*⟩

lemma (**in** *normed-vectorspace*) *fn-norm-ub* [*iff?*]:
  includes *fn-norm* + *continuous*
  assumes b: $b \in B\ V f$
  shows $b \leq \|f\|$-V
⟨*proof*⟩

lemma (**in** *normed-vectorspace*) *fn-norm-leastB*:
  includes *fn-norm* + *continuous*
  assumes b: $\bigwedge b.\ b \in B\ V f \Longrightarrow b \leq y$
  shows $\|f\|$-$V \leq y$
⟨*proof*⟩

The norm of a continuous function is always ≥ 0.

lemma (**in** *normed-vectorspace*) *fn-norm-ge-zero* [*iff*]:
  includes *fn-norm* + *continuous*
  shows $0 \leq \|f\|$-V
⟨*proof*⟩

The fundamental property of function norms is:

$$|f\,x| \leq \|f\| \cdot \|x\|$$

lemma (**in** *normed-vectorspace*) *fn-norm-le-cong*:
  includes *fn-norm* + *continuous* + *linearform*
  assumes x: $x \in V$
  shows $|f\,x| \leq \|f\|$-$V * \|x\|$
⟨*proof*⟩

The function norm is the least positive real number for which the following inequation holds:

$$|f\,x| \leq c \cdot \|x\|$$

lemma (**in** *normed-vectorspace*) *fn-norm-least* [*intro?*]:
  includes *fn-norm* + *continuous*
  assumes *ineq*: $\forall x \in V.\ |f\,x| \leq c * \|x\|$ **and** *ge*: $0 \leq c$
  shows $\|f\|$-$V \leq c$

⟨*proof*⟩

end

9 Zorn's Lemma

theory *ZornLemma* **imports** *Zorn* **begin**

Zorn's Lemmas states: if every linear ordered subset of an ordered set S has an upper bound in S, then there exists a maximal element in S. In our application, S is a set of sets ordered by set inclusion. Since the union of a chain of sets is an upper bound for all elements of the chain, the conditions of Zorn's lemma can be modified: if S is non-empty, it suffices to show that for every non-empty chain c in S the union of c also lies in S.

theorem *Zorn's-Lemma*:
  assumes r: $\bigwedge c.\ c \in \mathit{chain}\ S \Longrightarrow \exists\, x.\ x \in c \Longrightarrow \bigcup c \in S$
    and aS: $a \in S$
  shows $\exists\, y \in S.\ \forall\, z \in S.\ y \subseteq z \longrightarrow y = z$
⟨*proof*⟩

end

Part II
Lemmas for the Proof

10 The supremum w.r.t. the function order

theory *HahnBanachSupLemmas* **imports** *FunctionNorm ZornLemma* **begin**

This section contains some lemmas that will be used in the proof of the Hahn-Banach Theorem. In this section the following context is presumed. Let E be a real vector space with a seminorm p on E. F is a subspace of E and f a linear form on F. We consider a chain c of norm-preserving extensions of f, such that $\bigcup c = \mathit{graph}\ H\ h$. We will show some properties about the limit function h, i.e. the supremum of the chain c.

Let c be a chain of norm-preserving extensions of the function f and let *graph* H h be the supremum of c. Every element in H is member of one of the elements of the chain.

lemmas [*dest?*] = *chainD*
lemmas *chainE2* [*elim?*] = *chainD2* [*elim-format*, *standard*]

lemma *some-H'h't*:
  assumes M: $M = \mathit{norm\text{-}pres\text{-}extensions}\ E\ p\ F\ f$
    and cM: $c \in \mathit{chain}\ M$
    and u: $\mathit{graph}\ H\ h = \bigcup c$
    and x: $x \in H$
  shows $\exists H'\ h'.\ \mathit{graph}\ H'\ h' \in c$
    $\wedge\ (x,\ h\ x) \in \mathit{graph}\ H'\ h'$
    $\wedge\ \mathit{linearform}\ H'\ h' \wedge H' \trianglelefteq E$
    $\wedge\ F \trianglelefteq H' \wedge \mathit{graph}\ F\ f \subseteq \mathit{graph}\ H'\ h'$
    $\wedge\ (\forall x \in H'.\ h'\ x \leq p\ x)$
⟨*proof*⟩

Let c be a chain of norm-preserving extensions of the function f and let *graph* H h be the supremum of c. Every element in the domain H of the supremum function is member of the domain H' of some function h', such that h extends h'.

lemma *some-H'h'*:
  assumes M: $M = \mathit{norm\text{-}pres\text{-}extensions}\ E\ p\ F\ f$
    and cM: $c \in \mathit{chain}\ M$
    and u: $\mathit{graph}\ H\ h = \bigcup c$
    and x: $x \in H$
  shows $\exists H'\ h'.\ x \in H' \wedge \mathit{graph}\ H'\ h' \subseteq \mathit{graph}\ H\ h$
    $\wedge\ \mathit{linearform}\ H'\ h' \wedge H' \trianglelefteq E \wedge F \trianglelefteq H'$
    $\wedge\ \mathit{graph}\ F\ f \subseteq \mathit{graph}\ H'\ h' \wedge (\forall x \in H'.\ h'\ x \leq p\ x)$
⟨*proof*⟩

Any two elements x and y in the domain H of the supremum function h are both in the domain H' of some function h', such that h extends h'.

lemma *some-H'h'2*:

assumes *M*: $M = \textit{norm-pres-extensions}\ E\ p\ F\ f$
and *cM*: $c \in \textit{chain}\ M$
and *u*: $\textit{graph}\ H\ h = \bigcup c$
and *x*: $x \in H$
and *y*: $y \in H$
shows $\exists H'\ h'.\ x \in H' \wedge y \in H'$
$\wedge\ \textit{graph}\ H'\ h' \subseteq \textit{graph}\ H\ h$
$\wedge\ \textit{linearform}\ H'\ h' \wedge H' \trianglelefteq E \wedge F \trianglelefteq H'$
$\wedge\ \textit{graph}\ F\ f \subseteq \textit{graph}\ H'\ h' \wedge (\forall x \in H'.\ h'\ x \le p\ x)$
⟨*proof*⟩

The relation induced by the graph of the supremum of a chain c is definite, i. e. t is the graph of a function.

lemma *sup-definite*:
assumes *M-def*: $M \equiv \textit{norm-pres-extensions}\ E\ p\ F\ f$
and *cM*: $c \in \textit{chain}\ M$
and *xy*: $(x, y) \in \bigcup c$
and *xz*: $(x, z) \in \bigcup c$
shows $z = y$
⟨*proof*⟩

The limit function h is linear. Every element x in the domain of h is in the domain of a function h' in the chain of norm preserving extensions. Furthermore, h is an extension of h' so the function values of x are identical for h' and h. Finally, the function h' is linear by construction of M.

lemma *sup-lf*:
assumes *M*: $M = \textit{norm-pres-extensions}\ E\ p\ F\ f$
and *cM*: $c \in \textit{chain}\ M$
and *u*: $\textit{graph}\ H\ h = \bigcup c$
shows $\textit{linearform}\ H\ h$
⟨*proof*⟩

The limit of a non-empty chain of norm preserving extensions of f is an extension of f, since every element of the chain is an extension of f and the supremum is an extension for every element of the chain.

lemma *sup-ext*:
assumes *graph*: $\textit{graph}\ H\ h = \bigcup c$
and *M*: $M = \textit{norm-pres-extensions}\ E\ p\ F\ f$
and *cM*: $c \in \textit{chain}\ M$
and *ex*: $\exists x.\ x \in c$
shows $\textit{graph}\ F\ f \subseteq \textit{graph}\ H\ h$
⟨*proof*⟩

The domain H of the limit function is a superspace of F, since F is a subset of H. The existence of the 0 element in F and the closure properties follow from the fact that F is a vector space.

lemma *sup-supF*:
assumes *graph*: $\textit{graph}\ H\ h = \bigcup c$
and *M*: $M = \textit{norm-pres-extensions}\ E\ p\ F\ f$
and *cM*: $c \in \textit{chain}\ M$

and *ex*: $\exists x.\ x \in c$
and *FE*: $F \trianglelefteq E$
shows $F \trianglelefteq H$
⟨*proof*⟩

The domain H of the limit function is a subspace of E.

lemma *sup-subE*:
assumes *graph*: $\mathit{graph}\ H\ h = \bigcup c$
and *M*: $M = \mathit{norm\text{-}pres\text{-}extensions}\ E\ p\ F\ f$
and *cM*: $c \in \mathit{chain}\ M$
and *ex*: $\exists x.\ x \in c$
and *FE*: $F \trianglelefteq E$
and *E*: *vectorspace E*
shows $H \trianglelefteq E$
⟨*proof*⟩

The limit function is bounded by the norm p as well, since all elements in the chain are bounded by p.

lemma *sup-norm-pres*:
assumes *graph*: $\mathit{graph}\ H\ h = \bigcup c$
and *M*: $M = \mathit{norm\text{-}pres\text{-}extensions}\ E\ p\ F\ f$
and *cM*: $c \in \mathit{chain}\ M$
shows $\forall x \in H.\ h\ x \le p\ x$
⟨*proof*⟩

The following lemma is a property of linear forms on real vector spaces. It will be used for the lemma *abs-HahnBanach* (see page 23). For real vector spaces the following inequations are equivalent:

$$\forall x \in H.\ |h\ x| \le p\ x \quad \text{and} \quad \forall x \in H.\ h\ x \le p\ x$$

lemma *abs-ineq-iff*:
includes *subspace H E* + *vectorspace E* + *seminorm E p* + *linearform H h*
shows $(\forall x \in H.\ |h\ x| \le p\ x) = (\forall x \in H.\ h\ x \le p\ x)$ (**is** *?L = ?R*)
⟨*proof*⟩

end

11 Extending non-maximal functions

theory *HahnBanachExtLemmas* **imports** *FunctionNorm* **begin**

In this section the following context is presumed. Let E be a real vector space with a seminorm q on E. F is a subspace of E and f a linear function on F. We consider a subspace H of E that is a superspace of F and a linear form h on H. H is a not equal to E and x_0 is an element in $E - H$. H is extended to the direct sum $H' = H + \mathit{lin}\ x_0$, so for any $x \in H'$ the decomposition of $x = y + a \cdot x$ with $y \in H$ is unique. h' is defined on H' by $h'\ x = h\ y + a \cdot \xi$ for a certain ξ.

Subsequently we show some properties of this extension h' of h.

This lemma will be used to show the existence of a linear extension of f (see page **??**). It is a consequence of the completeness of $\mathbb{R}$. To show

$$\exists \xi.\ \forall y \in F.\ a\ y \leq \xi \wedge \xi \leq b\ y$$

it suffices to show that

$$\forall u \in F.\ \forall v \in F.\ a\ u \leq b\ v$$

lemma *ex-xi*:
  includes *vectorspace F*
  assumes r: $\bigwedge u\ v.\ u \in F \Longrightarrow v \in F \Longrightarrow a\ u \leq b\ v$
  shows $\exists\, xi{::}real.\ \forall y \in F.\ a\ y \leq xi \wedge xi \leq b\ y$
⟨*proof*⟩

The function h' is defined as a $h'\ x = h\ y + a \cdot \xi$ where $x = y + a \cdot \xi$ is a linear extension of h to H'.

lemma h'-*lf*:
  includes *var H* + *var h* + *var E*
  assumes h'-*def*: $h' \equiv \lambda x.\ let\ (y,\ a) =$
    $SOME\ (y,\ a).\ x = y + a \cdot x0 \wedge y \in H\ in\ h\ y + a * xi$
   and H'-*def*: $H' \equiv H + lin\ x0$
   and *HE*: $H \trianglelefteq E$
  includes *linearform H h*
  assumes *x0*: $x0 \notin H\ \ x0 \in E\ \ x0 \neq 0$
  includes *vectorspace E*
  shows *linearform* $H'\ h'$
⟨*proof*⟩

The linear extension h' of h is bounded by the seminorm p.

lemma h'-*norm-pres*:
  includes *var H* + *var h* + *var E*
  assumes h'-*def*: $h' \equiv \lambda x.\ let\ (y,\ a) =$
    $SOME\ (y,\ a).\ x = y + a \cdot x0 \wedge y \in H\ in\ h\ y + a * xi$
   and H'-*def*: $H' \equiv H + lin\ x0$
   and *x0*: $x0 \notin H\ \ x0 \in E\ \ x0 \neq 0$
  includes *vectorspace E* + *subspace H E* + *seminorm E p* + *linearform H h*
  assumes *a*: $\forall y \in H.\ h\ y \leq p\ y$
   and a': $\forall y \in H.\ -\ p\ (y + x0) - h\ y \leq xi \wedge xi \leq p\ (y + x0) - h\ y$
  shows $\forall x \in H'.\ h'\ x \leq p\ x$
⟨*proof*⟩

end

Part III

The Main Proof

12 The Hahn-Banach Theorem

theory *HahnBanach* **imports** *HahnBanachLemmas* **begin**

We present the proof of two different versions of the Hahn-Banach Theorem, closely following [1, §36].

12.1 The Hahn-Banach Theorem for vector spaces

Hahn-Banach Theorem. Let F be a subspace of a real vector space E, let p be a semi-norm on E, and f be a linear form defined on F such that f is bounded by p, i.e. $\forall\, x \in F.\ f\ x \leq p\ x$. Then f can be extended to a linear form h on E such that h is norm-preserving, i.e. h is also bounded by p.

Proof Sketch.

1. Define M as the set of norm-preserving extensions of f to subspaces of E. The linear forms in M are ordered by domain extension.
2. We show that every non-empty chain in M has an upper bound in M.
3. With Zorn's Lemma we conclude that there is a maximal function g in M.
4. The domain H of g is the whole space E, as shown by classical contradiction:
   - Assuming g is not defined on whole E, it can still be extended in a norm-preserving way to a super-space H' of H.
   - Thus g can not be maximal. Contradiction!

theorem *HahnBanach*:
  includes *vectorspace* E + *subspace* F E + *seminorm* E p + *linearform* F f
  assumes *fp*: $\forall\, x \in F.\ f\ x \leq p\ x$
  shows $\exists\, h.$ *linearform* $E\ h \wedge (\forall\, x \in F.\ h\ x = f\ x) \wedge (\forall\, x \in E.\ h\ x \leq p\ x)$
  — Let E be a vector space, F a subspace of E, p a seminorm on E,
  — and f a linear form on F such that f is bounded by p,
  — then f can be extended to a linear form h on E in a norm-preserving way.
⟨*proof*⟩

12.2 Alternative formulation

The following alternative formulation of the Hahn-Banach Theorem uses the fact that for a real linear form f and a seminorm p the following inequations are equivalent:[1]

$$\forall\, x \in H.\ |h\ x| \leq p\ x \quad \text{and} \quad \forall\, x \in H.\ h\ x \leq p\ x$$

[1] This was shown in lemma *abs-ineq-iff* (see page 21).

theorem *abs-HahnBanach*:
includes *vectorspace* E + *subspace* F E + *linearform* F f + *seminorm* E p
assumes *fp*: $\forall x \in F.\ |f\ x| \le p\ x$
shows $\exists g.$ *linearform* E g
$\wedge\ (\forall x \in F.\ g\ x = f\ x)$
$\wedge\ (\forall x \in E.\ |g\ x| \le p\ x)$
⟨*proof*⟩

12.3 The Hahn-Banach Theorem for normed spaces

Every continuous linear form f on a subspace F of a norm space E, can be extended to a continuous linear form g on E such that $\|f\| = \|g\|$.

theorem *norm-HahnBanach*:
includes *normed-vectorspace* E + *subspace* F E + *linearform* F f + *fn-norm* + *continuous* F *norm* $(\|\text{-}\|)$ f
shows $\exists g.$ *linearform* E g
$\wedge$ *continuous* E *norm* g
$\wedge\ (\forall x \in F.\ g\ x = f\ x)$
$\wedge\ \|g\|\text{-}E = \|f\|\text{-}F$
⟨*proof*⟩

end

References

[1] H. Heuser. *Funktionalanalysis: Theorie und Anwendung*. Teubner, 1986.

[2] L. Narici and E. Beckenstein. The Hahn-Banach Theorem: The life and times. In *Topology Atlas*. York University, Toronto, Ontario, Canada, 1996. http://at.yorku.ca/topology/preprint.htm and http://at.yorku.ca/p/a/a/a/16.htm.

[3] B. Nowak and A. Trybulec. Hahn-Banach theorem. *Journal of Formalized Mathematics*, 5, 1993. http://mizar.uwb.edu.pl/JFM/Vol5/hahnban.html.