Are All Triangles Equilateral?

Start with an arbitrary triangle $\triangle ABC$ in the plane.
Are All Triangles Equilateral?

Draw a line which bisects the angle at C.
The perpendicular bisector of $\overline{AB}$ intersects that line in a point $M$. 

Are All Triangles Equilateral?
Are All Triangles Equilateral?

Draw from the intersection $M$ the normal lines to the other two sides.
Finally, connect the point $M$ to $A$ and $M$ to $B$. 
Are All Triangles Equilateral?

The right triangles $RMC$ and $MQC$ have a common side $MC$ and the same angle at $C$. 
Therefore, the line segments $QC$ and $RC$ have the same length.
The right triangles $\triangle APM$ and $\triangle PBM$ are congruent because they have 2 equal sides.
Therefore the segments AM and BM are the same length.
Are All Triangles Equilateral?

The two right triangles AMR and BQM are congruent because they have two equal sides.
Therefore the segments $AR$ and $BQ$ have equal length.
Are All Triangles Equilateral?

Since \|AC\| = \|AR\| + \|RC\| = \|BQ\| + \|QC\| = \|BC\|

the triangle \(ABC\) is isosceles.

The same argument holds for \(|AB| = |AC|\)
Are All Triangles Equilateral?

Therefore, all triangles are equilateral.
What is a Proof?

• **Diagrams** can be a minefield for *mistakes*

• So what is a proof?
  – One which is *accepted* by community?
  – Human *intuition* needed?
  – Completely logical?
Axiomatic Approach

• Axioms are assumptions of a universe

• Axioms combined with rules of logic infer new theorems

• Hilbert's *Grundlagen der Geometrie* followed this approach
  - published in 1899
  - rigorous axiomatisation of Euclidean space
Hilbert's Grundlagen

- 3 **primitive objects**: points, lines, planes
  - **Claim**: it is not necessary to assign any explicit meaning to these primitives
  - They could be chairs, tables and beer mugs!
- Relationships between the primitives described and categorised into 5 groups of axioms
  - Using **primitive relations**: on line, between, ...
  - Axioms **minimal** and **complete**
  - Ex: *for every two points A, B there exists a line a that contains each of the points A, B.*

Hilbert claimed his proofs were *free of intuition* and *required only his axioms and the rules of logic*
Theorem Three

Theorem three: $A \neq C \Rightarrow \exists D.$ between $A$ $D$ $C$
Theorem Three

**Grundlagen Proof:**

By Axiom (I,3) there exists a point $E$ outside the line $AC$.

**Missing:**

Need to construct a line that $A$ and $C$ lie on.
Theorem Three

Grundlagen Proof:

By Axiom (II,2) there exists on $AE$ a point $F$ such that $E$ is on the segment $AF$.

Missing:
Need to show $A$ and $E$ are distinct.
Theorem Three

Grundlagen Proof:

By Ax (II,2) and Ax (II,3) there exists on FC a point G that does not lie on the segment FC.

Missing:
Need to show F and C are distinct.
Theorem Three

\[ \text{AxII4: } \neg \text{coll}\{A,B,C\} \land \text{lineOnPlane a (planeOf A B C)} \land \text{lineMeetsSeg a A B } \land \neg \text{onLine A a } \land \neg \text{onLine B a } \land \neg \text{onLine C a } \Rightarrow (\text{lineMeetsSeg a A C } \lor \text{lineMeetsSeg a B C}) \]

Grundlagen Proof:

By Ax (II,4) the line EG must then intersect the line AC at a point D.

Missing: Need to show points and lines planar, case split EG intersects FC is a contradiction.
Observations

• Hilbert made implicit assumptions
  - newly constructed points were distinct
  - the existence of specific lines (i.e. AC)
  - all points and lines were planar
  - case split omitted

• Diagram appeals to our intuition

• Diagram could be reason for missing steps in proof
Story So Far ... 

- Proving geometric results is challenging:
  - **Diagrams** can be misleading
  - Even Hilbert relied on intuition
- Confidence in geometric results suspect?
- Formal computerised proof would give reassurances
  - especially needed when results relied upon for safety-critical applications
Computational Geometry

- Databases
- Computer graphics
- Computer vision
- Molecular biology
- Manufacturing
- Robotics
- Statistics
- Air Traffic Control
Convex Hull Problem
Convex Hull Problem
Convex Hull Problem
Formal Spec. of Convex Hull

The convex hull of a set of planar points Q is:

1. The intersection of all convex sets that contain Q
2. The union of all the triangles determined by points in Q
3. The set of all convex combinations of the points of Q

... many more definitions

Which definition is best to formalise? Let's first consider the algorithm we are going to verify.
Graham’s Scan

• Graham’s Scan computes 2D convex hull
• Input is set of 3 or more distinct points
• Uses rotational sweep technique
  – points ordered using polar angle
• Maintains stack $S$ of candidate points
• Each point in $Q$ pushed on once
• Points which are not vertices are eventually popped
Graham’s Scan

Find rightmost lowest point; label it p0. Sort all other points angularly about p0, break ties in favour of closeness to p0; label p1, ..., pn-1.

Stack S=(pn-1,p0)=(pt-1,pt); t indexes top.

\[ i = 1 \]

while \( i < n \) do
  if \( p_i \) is strictly left of \((pt-1,pt)\) then Push(S,p_i) and set \( i \to i + 1 \)
  else Pop(S)

Graham’s Scan
Graham’s Scan

$S = \{p_1, p_0, p_{11}\}$
Graham’s Scan

S = \{p2, p1, p0, p11\}
Graham’s Scan

\[ S = [p3, p2, p1, p0, p11] \]
Graham’s Scan

S = [p2, p1, p0, p11]

Pop p3
Graham’s Scan

$S = \{p_4, p_2, p_1, p_0, p_{11}\}$
Graham’s Scan

\[ S = [p_{10}, p_{9}, p_{8}, p_{5}, p_{2}, p_{1}, p_{0}, p_{11}] \]
Graham’s Scan

S = [p11, p9, p8, p5, p2, p1, p0, p11]

Two of p11 on stack. Need to pop one
Formal Spec. of Convex Hull (II)

- **Left turn** important concept of alg.
- Knuth's “Axioms and Hulls” defines convex hull in terms of left turns
  - axiomatic approach
  - axioms determine a **counter-clockwise (CC) system**
  - \(tsp\) represents a left turn travelling from \(t\) to \(s\) to \(p\)
Knuth's Counter-Clockwise System

- **Ax 1** (cyclic symmetry). $pqr \Rightarrow qrp$
- **Ax 2** (antisymmetry). $pqr \Rightarrow \neg prq$
- **Ax 3** (nondegeneracy). $pqr \lor prq$
- **Ax 4** (interiority). $tqr \land ptr \land pqt \Rightarrow pqr$
- **Ax 5** (transitivity).

\[ tsp \land tsq \land tsr \land tpq \land tqr \]
\[ \Rightarrow tpr \]
Knuth’s Definition

The convex hull of a CC system $Q$ is the set of all ordered pairs $ts$ of distinct points such that $tsp$ holds for all $p$ in $Q$, $p$ not in $\{s,t\}$.
Knuth’s Definition

What if $p$ lies between 2 vertices?
Then $tsp$ does not hold for all $p$ in $Q$!
CC system excludes degenerate cases.
Extension to CC System

• To permit collinear points, notion of *betweenness* introduced

• Axioms updated to incorporate this change
Formal Spec. Convex Hull (III)

S \text{ isConvexHull } Q \equiv
\begin{align*}
\text{distinct } S \land \text{ set } S & \subseteq \text{ set } Q \land \\
( \forall n < \text{length } Q. \forall i < \text{length } S - 1. \\
& \quad ((S!i+1)(S!i)(Q!n) \lor \\
& \quad (Q!n) \text{ mem } [S!i+1, S!i] \lor \\
& \quad (Q!n) \text{ isBetween } (S!i+1) (S!i)) \\
& \land \\
& \quad ((\text{hd } S)(\text{last } S)(Q!n) \lor \\
& \quad (Q!n) \text{ mem } [\text{hd } S, \text{ last } S] \lor \\
& \quad (Q!n) \text{ isBetween } (\text{hd } S) (\text{last } S)) \}
\end{align*}
Floyd-Hoare Logic

• Logic for reasoning mathematically about imperative programs
• Used to verify imperative programs
• Partial correctness specification:

\[ \{P\} \quad C \quad \{Q\} \]

• Total correctness =
  Partial Correctness + Termination
Example

preconditions

\{X=x \land Y=y\}

BEGIN R := X; X := Y; Y := R; END

\{X=y \land Y=x\}

postconditions

program
Floyd-Hoare Logic (II)

• Partial correctness specification is annotated with mathematical statements called a loop invariant
  – loop invariant is the facts which remain true every time a loop is entered or left

• Verification conditions (VCs) are then produced by the logic

• VCs provable $\rightarrow$ specification correct
VCs for WHILE-command

\{P\} WHILE S DO \{R\} C \{Q\}

VCs to prove are

i. \(P \implies R\)

ii. \(R \land S \implies \text{body of loop preserves } R\)

iii. \(R \land \neg S \implies Q\)
Floyd-Hoare Logic in Isabelle

\{ P \}.

Initialize local variables

WHILE S

INV \{ R \}.

DO

C

OD

\{ Q \}.

VCs are automatically generated in Isabelle. These are statements in HOL which need to be proved.
Back to Graham’s Scan

\{ \text{ordered } Q \text{ & } 3 \leq \text{length } Q \text{ & } \text{distinct } Q \text{ & } \neg \text{all-collinear } Q \}.

\begin{align*}
\text{'}i&:=1; \\
\text{'}S&:=[\text{hd } Q, \text{last } Q]; \\
\text{WHILE } \text{'}i < \text{length } Q \\
\text{INV } \{ 2 \leq \text{length } \text{'}S \text{ & } \\
&\quad \text{'}i \leq \text{length } Q \text{ & } \\
&\quad \text{distinct } Q \text{ & } \\
&\quad \text{distinct } (\text{butlast } \text{'}S) \text{ & } \\
&\quad \text{..... } \}
\end{align*}

\text{DO}
\begin{align*}
\text{IF } &\text{Left-turn } (\text{'}S \text{ ! 1}) (\text{'}S \text{ ! 0}) (Q \text{ ! } \text{'}i) \\
\text{THEN } &\text{'}S := (Q \text{ ! } \text{'}i) \# \text{'}S; \\
&\quad \text{'}i := \text{'}i + 1 \\
\text{ELSE } &\text{'}S := (\text{tl } \text{'}S) \\
\text{FI}
\end{align*}
\text{OD}
\begin{align*}
\{ \text{butlast } \text{'}S \text{ isConvexHull } Q \}.
\end{align*}

we have abstracted the stack into a list
Loop Invariant

There are many components to the loop invariant. Three more components are:

\[
\forall j \ k \ l. \ (j < \text{length } 'S - 2 \land k < j \land l < k ) \rightarrow ( 'S ! j)( 'S ! k)( 'S ! l)
\]

\[
' i = \text{length } Q \rightarrow \text{last } Q = \text{hd } ' S
\]

\[
\forall k < \text{length } 'S-1. \ \exists n < \text{length } Q. \\
'S!k = Q!n \land \\
( (\text{drop } k (\text{butlast } 'S)) \text{ isConvexHull} (\text{take } (n+1) Q) \lor \\
((\text{all_collinear} (n+1) Q) \land \\
( \text{length } 'S - k = 2 \lor \text{length } 'S - k = 3)))
\]
Third VC Generated

**iii. R \land \neg S \Rightarrow Q**

\neg 'i < \text{length } Q \land 'i \leq \text{length } Q \land 
\neg (i = \text{length } Q \rightarrow \text{last } Q = \text{hd } 'S) \land 
\neg \text{all-collinear } Q \land 
( \forall k < \text{length } 'S-1. \exists n < \text{length } Q. 'S!k = Q!n \land 
( \text{drop } k \text{ (butlast } 'S)) \text{ isConvexHull (take } (n+1) \text{ Q) } \lor 
(\text{all_collinear } (n+1) \text{ Q) } \land 
( \text{length } 'S - k = 2 \lor \text{length } 'S - k = 3))) 
\rightarrow \text{butlast } 'S \text{ isConvexHull } Q

From assumptions 'i must be equal to length Q
Third VC Generated

iii. \( R \land \neg S \Rightarrow Q \)

\('i = \text{length } Q \land \n
('i = \text{length } Q \rightarrow \text{last } Q = \text{hd } S) \land \n
\neg \text{all-collinear } Q \land \n
( \forall k < \text{length } S-1. \exists n < \text{length } Q. \ 'S!k = Q!n \land \n
( (\text{drop } k (\text{butlast } S)) \text{ isConvexHull } (\text{take } (n+1) Q) \lor \n
( (\text{all_collinear } (n+1) Q) \land \n
( \text{length } S - k = 2 \lor \text{length } S - k = 3) ) \) ) \n
\rightarrow (\text{butlast } S) \text{ isConvexHull } Q \n
Can then infer: \( \text{last } Q = \text{hd } S, \)
and instantiate: \( k = 0 \)
Third VC Generated

### iii. $R \land \neg S \Rightarrow Q$

\[
\text{last } Q = \text{hd 'S } \land \\
\neg \text{all-collinear } Q \land \\
( \exists \ n < \text{length } Q. \ 'S!0 = Q!n \land \\
(\ (\text{drop } k \ (\text{butlast 'S})) \ \text{isConvexHull} \ (\text{take } (n+1) \ Q) \ \lor \\
(\ (\text{all_collinear} \ (n+1) \ Q) \ \land \\
(\ \text{length 'S} - k = 2 \ \lor \ \text{length 'S} - k = 3)) ) \\
\rightarrow (\text{butlast 'S}) \ \text{isConvexHull} \ Q
\]

We then deduce that $n = \text{length } Q - 1$
Third VC Generated

iii. \[ R \land \neg S \Rightarrow Q \]

\[ \neg \text{all-collinear } Q \land (\text{butlast 'S) isConvexHull } Q \lor ((\text{all_collinear } Q) \land (\text{length 'S - k = 2 } \lor \text{length 'S - k = 3})) \rightarrow (\text{butlast 'S) isConvexHull } Q \]

We then get a case split:
1\textsuperscript{st} case implies conclusion, 2\textsuperscript{nd} case is a contradiction.
Remarks on Proof

- Discovering correct loop invariant is:
  - difficult, iterative process of refining
  - hindered due to Emacs PG's poor support for re-factoring

- Writing own tactics/automation is challenging
  - can be aided by FeaSch-on-Isabelle

- Alternative to axiomatic approach?
  - Isabelle methodology prefers theories to be conservative extensions of the library
  - We could define left turn!
Could build on theory of 2D real vectors.

To capture the notion of a *left turn*:

\[
tsp \Rightarrow (s - t) \llp > (p - t) > 0
\]

where \( \llp > \) is defined as the outer product:

\[
P \llp > Q \equiv P_x Q_y - P_y Q_x
\]
typedef realv = "\{p :: (real * real). True\}"

instance
realv :: \{zero, plus, minus\}

consts
"<*>" :: [realv, realv] \rightarrow real

defs
realv_oprod_def
"P <*> Q \equiv ((p1, p2), (q1, q2)). p1 * q2 - p2 * q1
(Rep_realv(P), Rep_realv(Q))"
Betweenness

- betweenness can be defined as:

\[ p \text{ isBetween } t s \equiv \]
\[ (s - t) \leftrightarrow (p - t) = 0 \land \]
\[ |pt| < |ts| \land \]
\[ |ps| < |ts| \]

- Does the definitional approach complicate the proof?
Proving Knuth's Axiom 5

Axiom 5. \( tsp \land tsq \land tsr \land tpq \land tqr \Rightarrow tpr \)

- Proof breaks down into:
  - non-linear equations, difficult to solve
  - many case splits and tedious computation
- How can we ease the proving process?
Real Algebra

- Decidability for the first order theory of real closed fields is most fundamental result with respect to real numbers (shown by Tarski)
- Collins gave first practical decision algorithm for this problem
- However, no decision procedure within Isabelle
- But, QEPCAD can help:
  - CAD (Cylindrical Algebraic Decomposition)
  - QEPCAD also gives a method for QE (Quantifier Elimination)
Proof Engineering Approach to Systems Integration

- USER INTERFACE
- Eclipse Proof General
- TACTICS
  - Feasch-on-Isabelle
- COMPUTER ALGEBRA
  - QEPCAD
- THEOREM PROVER
  - Isabelle
Benefits of PE Approach

• Modularity and interoperability:
  – QEPCAD widget could work standalone
  – result available for many systems

• User has more control
  – setting parameters
  – changing translations (input and output)

• Greater inspectability