Topics: reductions as a way of showing how incomputability “spreads” from the halting function to other important functions; then, a return to an upside of impossibility: zero-knowledge protocols.

I. Reminder: the halting function \( M_i \) denotes the \( i \)th B-input TM.

\[
h(M_i, j) = \begin{cases} 
1 \text{ (i.e., yes)}, & \text{if } M_i \text{ would halt given } j \text{ B’s as input} \\
0 \text{ (i.e., no)}, & \text{if } M_i \text{ would not halt given } j \text{ B’s as input}
\end{cases}
\]

II. The “at most 2 hang-inducing inputs” function

\[
f_{\leq 2 \text{ hangers}}(M_k) = \begin{cases} 
1 \text{ (i.e., yes)}, & \text{if there are at most 2 inputs (sequences of B’s) on which } M_k \text{ doesn’t halt} \\
0 \text{ (i.e., no)}, & \text{if there are at least 3 inputs on which } M_k \text{ runs forever}
\end{cases}
\]

III. A Trojan-horse program Given a B-input TM \( M_i \) and a number \( j \) (corresponding to an input to \( M_i \)), we can construct the program for a B-input TM \( T_{M_i,j} \) that acts as follows:

Given \( \ell \) B’s as input,
- if \( \ell \neq 13 \) and \( \ell \neq 666 \) and \( \ell \neq 172 \), halt immediately;
- otherwise (i.e., \( \ell = 13 \) or \( \ell = 666 \) or \( \ell = 172 \)), run \( M_i \) on \( j \) B’s as input.

IV. The “at most a finite number of hang-inducing inputs” function

\[
f_{\text{finite hangers}}(M_k) = \begin{cases} 
1 \text{ (i.e., yes)}, & \text{if there are only a finite number of inputs on which } M_k \text{ doesn’t halt} \\
0 \text{ (i.e., no)}, & \text{if there are an infinite number of inputs on which } M_k \text{ runs forever}
\end{cases}
\]

V. 3-colorability A graph (collection of nodes and edges between some pairs of nodes) is 3-colorable if one can, using at most three colors, assign each node a color in such a way that for each edge in the graph, the two nodes on the edge’s endpoints have different colors.

VI. “Zero knowledge” protocol for 3-colorability

You declare your three colors (e.g., red, green, blue).
(*) Off-stage, randomly permute your coloring (e.g., red ↔ green, blue stays the same).
Present color-hidden graph.
Suspicious entity chooses an edge.
You reveal the edge’s two endpoints (they ought to be different colors, and from your declared set).