Thm 4.11 A_{TM} is undecidable.

<u>proof</u>: We prove this theorem by contradiction.

Assume H is a decider for A_{TM},

$$\underset{(\text{on} \langle M, w \rangle)}{H} = \begin{cases} \text{accept}, & \text{if } M \text{ accepts } w \\ \text{reject}, & \text{if } M \text{ doesn't accept } w \end{cases}$$

Then, construct D, with $\langle M \rangle$ as input

1. Run H on $\langle M, \langle M \rangle \rangle$.
2. output the opposite of what H outputs.

Therefore,

$$D(\langle M \rangle) = \begin{cases} \text{accept}, & \text{if } M \text{ doesn't accept } \langle M \rangle. \\ \text{reject}, & \text{if } M \text{ accepts } \langle M \rangle. \end{cases}$$

Finally, we feed $\langle D \rangle$ into D to obtain the contradiction.

$$D(\langle D \rangle) = \begin{cases} \text{accept}, & \text{if } D \text{ doesn't accept } \langle D \rangle. \\ \text{reject}, & \text{if } D \text{ accepts } \langle D \rangle. \end{cases}$$

$\therefore$ A_{TM} is undecidable. $\square$

Does this have anything to do with the diagonalization method?

	$\langle M_1 \rangle$	$\langle M_2 \rangle$	$\langle M_3 \rangle$	$\cdots$
M_1	accept		accept	
M_2	accept	accept	accept	
M_3	accept			
$\vdots$				

M_i accept $\langle M_j \rangle$

	$\langle M_1 \rangle$	$\langle M_2 \rangle$	$\langle M_3 \rangle$
M_1	accept	reject	accept
M_2	accept	accept	accept
M_3	accept	reject	reject
$\vdots$			

Run H on $\langle M_i, \langle M_j \rangle \rangle$

	$\langle M_1 \rangle$	$\langle M_2 \rangle$	$\langle M_3 \rangle$	$\cdots$	$\langle D \rangle$
M_1	accept	reject	accept	$\cdots$	
M_2	accept	accept	accept	$\cdots$	
M_3	accept	reject	reject	$\cdots$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$		
D					?

D causes a contradiction along the diagonal

D rejects $\langle D \rangle$ exactly when D accepts $\langle D \rangle$

- If proving sth to be undecidable is so hard, this shouldn't be taught at undergraduate level.
- There is a simple way, relatively speaking, to do this.
- <u>Reduction</u>: $A \propto B$ ($A \le B$), if A can be solved using B as a subroutine.

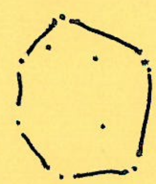
// Notice the order, 90% of people will get it wrong, at the first try!

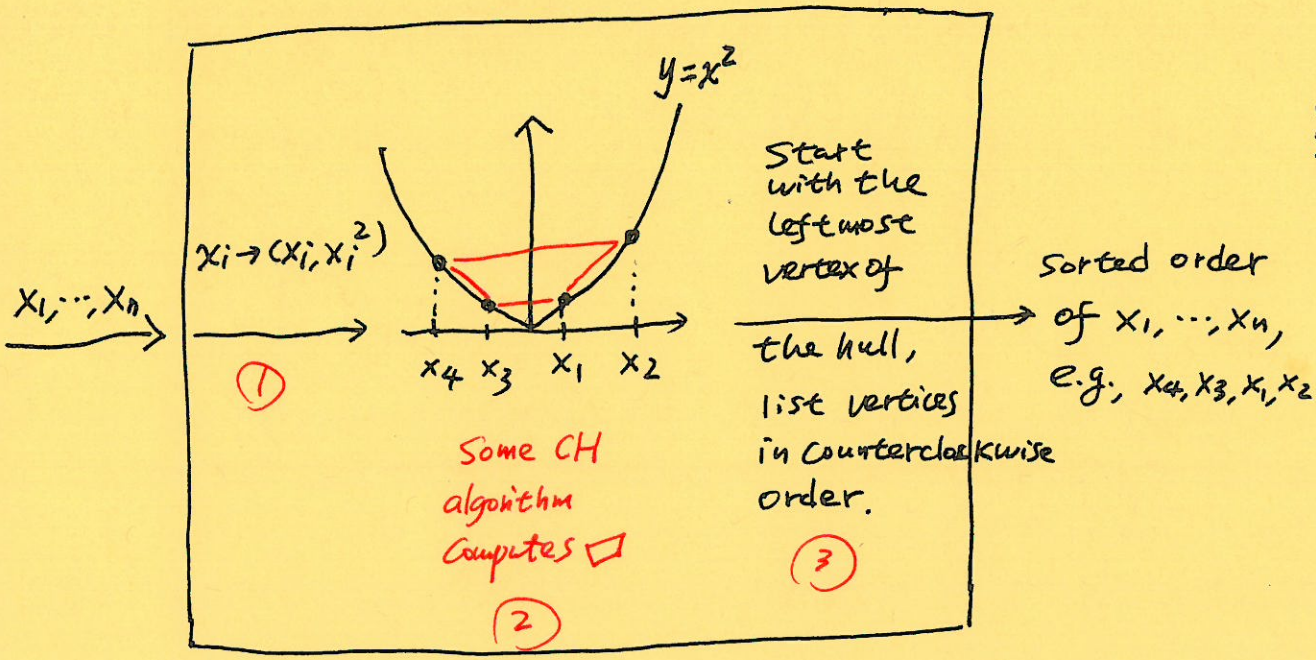
- If we know in advance that A is hard, then B is at least as hard as A.

Sorting $\le$ 2D Convex hull

(n reals $x_1, x_2, \cdots, x_n$)

n points in 2D





We know sorting takes $\Omega(n \log n)$ time, i.e.,

① + ② + ③ is at least $\Omega(n \log n)$.

But ①, ③ can be done in $O(n)$ time, so ② must take $\Omega(n \log n)$ time, or <u>2D convex hull</u> takes $\Omega(n \log n)$ time!