Computability Theory: Decidability and Recognizability

- Encoding Turing Machines and the Universal TM
- Computability
- Halt: Undecidable Problems using Diagnolization
- Accept: Undecidable Problems using Diagnolization
- Turing Reductions
- Mapping Reductions
- Undecidable and Unrecognizable Problems
- Rice Theorem

Imdad ullah Khan

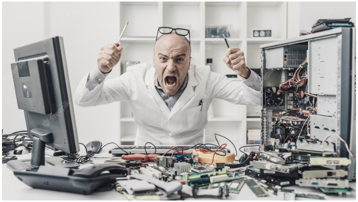
The ACCEPT Problem

Another concrete Undecidable Problem

$A_{TM} = \{\langle M, w\rangle : M \text{ a TM on } \Sigma,\ M \text{ accepts } w \in \Sigma^*\}$

Why would you be interested in A_{TM}? Can you use it for autograding?

[Turing 1936] A_{TM} is recognizable, but not decidable



A_{TM} is undecidable

$A_{TM} = \{\langle M, w\rangle : M \text{ a TM on } \Sigma,\ M \text{ accepts } w \in \Sigma^*\}$ is undecidable

Proof: Suppose A_{TM} is decidable i.e. there exists a TM M_A s.t.

$$M_A(\langle M, w\rangle) = \begin{cases} \textbf{Accept} & \text{if } M \text{ accepts } w \\ \textbf{Reject} & \text{if } M \text{ does not accept } w \end{cases}$$

Define a TM D that takes input encoding of a TM as follows:

$D(\langle M\rangle)$:

Run M_A on $\langle M, M\rangle$ and output the opposite of M_A

$$D(\langle M\rangle) = \begin{cases} \textbf{Reject} & \text{if } M_A \text{ accepts } \langle M, M\rangle \\ \textbf{Accept} & \text{if } M_A \text{ does not accept } \langle M, M\rangle \end{cases}$$

- M_A outputs the same as its input M on any string w
- D outputs the opposite of its input M on the fixed string $w = \langle M\rangle$

What if we run D on $\langle D\rangle$

$$D(\langle D\rangle) = \begin{cases} \textbf{Reject} & \text{if } D \text{ accepts } \langle D\rangle \\ \textbf{Accept} & \text{if } D \text{ does not accept } \langle D\rangle \end{cases}$$

A_{TM} is undecidable

$A_{TM} = \{\langle M, w\rangle : M \text{ a TM on } \Sigma,\ M \text{ accepts } w \in \Sigma^*\}$ is undecidable

Proof: Suppose A_{TM} is decidable, i.e. there exists a TM M_A

$$M_A(\langle M, w\rangle) = \begin{cases} \textbf{Accept} & \text{if } M \text{ accepts } w \\ \textbf{Reject} & \text{if } M \text{ doesn't accept } w \end{cases}$$

$D(\langle M\rangle)$: Run M_A on $\langle M, M\rangle$ and output the opposite of M_A

$$D(\langle M\rangle) = \begin{cases} \textbf{Reject} & \text{if } M \text{ accepts } \langle M\rangle \\ \textbf{Accept} & \text{if } M \text{ doesn't accept } \langle M\rangle \end{cases}$$

Partial Output Table of $M_A(\langle M, w\rangle)$

	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\langle M_5\rangle$	$\cdots$	$\langle D\rangle$	$\cdots$
M_1	accept	accept	reject	reject	reject		accept	
M_2								
M_3								
M_4								
M_5								
$\vdots$								
D								
$\vdots$								

A_{TM} is undecidable

$A_{TM} = \{\langle M, w\rangle : M \text{ a TM on } \Sigma,\ M \text{ accepts } w \in \Sigma^*\}$ is undecidable

Proof: Suppose A_{TM} is decidable, i.e. there exists a TM M_A

$D(\langle M\rangle)$: Run M_A on $\langle M, M\rangle$ and output the opposite of M_A

$$M_A(\langle M, w\rangle) = \begin{cases} \textbf{Accept} & \text{if } M \text{ accepts } w \\ \textbf{Reject} & \text{if } M \text{ doesn't accept } w \end{cases}$$

$$D(\langle M\rangle) = \begin{cases} \textbf{Reject} & \text{if } M \text{ accepts } \langle M\rangle \\ \textbf{Accept} & \text{if } M \text{ doesn't accept } \langle M\rangle \end{cases}$$

Partial Output Table of $M_A(\langle M, w\rangle)$

	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\langle M_5\rangle$	$\cdots$	$\langle D\rangle$	$\cdots$
M_1	**accept**	**accept**	**reject**	**reject**	**reject**		**accept**	
M_2	**accept**	**reject**	**reject**	**accept**	**reject**		**accept**	
M_3	**reject**	**accept**	**accept**	**reject**	**accept**		**reject**	
M_4	**accept**	**reject**	**accept**	**accept**	**reject**		**reject**	
M_5	**reject**	**accept**	**reject**	**accept**	**reject**		**accept**	
$\vdots$								
D								
$\vdots$								

A_{TM} is undecidable

$A_{TM} = \{\langle M, w\rangle : M \text{ a TM on } \Sigma,\ M \text{ accepts } w \in \Sigma^*\}$ is undecidable

Proof: Suppose A_{TM} is decidable, i.e. there exists a TM M_A

$D(\langle M\rangle)$: Run M_A on $\langle M, M\rangle$ and output the opposite of M_A

$$M_A(\langle M, w\rangle) = \begin{cases} \textbf{Accept} & \text{if } M \text{ accepts } w \\ \textbf{Reject} & \text{if } M \text{ doesn't accept } w \end{cases}$$

$$D(\langle M\rangle) = \begin{cases} \textbf{Reject} & \text{if } M \text{ accepts } \langle M\rangle \\ \textbf{Accept} & \text{if } M \text{ doesn't accept } \langle M\rangle \end{cases}$$

Partial Output Table of $M_A(\langle M, w\rangle)$

	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\langle M_5\rangle$	$\cdots$	$\langle D\rangle$	$\cdots$
M_1	**accept**	**accept**	**reject**	**reject**	**reject**		**accept**	
M_2	**accept**	**reject**	**reject**	**accept**	**reject**		**accept**	
M_3	**reject**	**accept**	**accept**	**reject**	**accept**		**reject**	
M_4	**accept**	**reject**	**accept**	**accept**	**reject**		**reject**	
M_5	**reject**	**accept**	**reject**	**accept**	**reject**		**accept**	
$\vdots$								
D	**reject**							
$\vdots$								

A_{TM} is undecidable

$A_{TM} = \{\langle M, w\rangle : M \text{ a TM on } \Sigma,\ M \text{ accepts } w \in \Sigma^*\}$ is undecidable

Proof: Suppose A_{TM} is decidable, i.e. there exists a TM M_A

$$M_A(\langle M, w\rangle) = \begin{cases} \textbf{Accept} & \text{if } M \text{ accepts } w \\ \textbf{Reject} & \text{if } M \text{ doesn't accept } w \end{cases}$$

$D(\langle M\rangle)$: Run M_A on $\langle M, M\rangle$ and output the opposite of M_A

$$D(\langle M\rangle) = \begin{cases} \textbf{Reject} & \text{if } M \text{ accepts } \langle M\rangle \\ \textbf{Accept} & \text{if } M \text{ doesn't accept } \langle M\rangle \end{cases}$$

Partial Output Table of $M_A(\langle M, w\rangle)$

	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\langle M_5\rangle$	$\cdots$	$\langle D\rangle$	$\cdots$
M_1	accept	accept	reject	reject	reject		accept	
M_2	accept	reject	reject	accept	reject		accept	
M_3	reject	accept	accept	reject	accept		reject	
M_4	accept	reject	accept	accept	reject		reject	
M_5	reject	accept	reject	accept	reject		accept	
$\vdots$								
D	reject	accept						
$\vdots$								

A_{TM} is undecidable

$A_{TM} = \{\langle M, w\rangle : M \text{ a TM on } \Sigma,\ M \text{ accepts } w \in \Sigma^*\}$ is undecidable

Proof: Suppose A_{TM} is decidable, i.e. there exists a TM M_A

$$M_A(\langle M, w\rangle) = \begin{cases} \textbf{Accept} & \text{if } M \text{ accepts } w \\ \textbf{Reject} & \text{if } M \text{ doesn't accept } w \end{cases}$$

$D(\langle M\rangle)$: Run M_A on $\langle M, M\rangle$ and output the opposite of M_A

$$D(\langle M\rangle) = \begin{cases} \textbf{Reject} & \text{if } M \text{ accepts } \langle M\rangle \\ \textbf{Accept} & \text{if } M \text{ doesn't accept } \langle M\rangle \end{cases}$$

Partial Output Table of $M_A(\langle M, w\rangle)$

	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\langle M_5\rangle$	$\cdots$	$\langle D\rangle$	$\cdots$
M_1	**accept**	**accept**	**reject**	**reject**	**reject**		**accept**	
M_2	**accept**	**reject**	**reject**	**accept**	**reject**		**accept**	
M_3	**reject**	**accept**	**accept**	**reject**	**accept**		**reject**	
M_4	**accept**	**reject**	**accept**	**accept**	**reject**		**reject**	
M_5	**reject**	**accept**	**reject**	**accept**	**reject**		**accept**	
$\vdots$								
D	**reject**	**accept**	**reject**	**reject**	**accept**			
$\vdots$								

A_{TM} is undecidable

$A_{TM} = \{\langle M, w\rangle : M \text{ a TM on } \Sigma,\ M \text{ accepts } w \in \Sigma^*\}$ is undecidable

Proof: Suppose A_{TM} is decidable, i.e. there exists a TM M_A

$$M_A(\langle M, w\rangle) = \begin{cases} \textbf{Accept} & \text{if } M \text{ accepts } w \\ \textbf{Reject} & \text{if } M \text{ doesn't accept } w \end{cases}$$

$D(\langle M\rangle)$: Run M_A on $\langle M, M\rangle$ and output the opposite of M_A

$$D(\langle M\rangle) = \begin{cases} \textbf{Reject} & \text{if } M \text{ accepts } \langle M\rangle \\ \textbf{Accept} & \text{if } M \text{ doesn't accept } \langle M\rangle \end{cases}$$

Partial Output Table of $M_A(\langle M, w\rangle)$

	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\langle M_5\rangle$	$\cdots$	$\langle D\rangle$	$\cdots$
M_1	accept	accept	reject	reject	reject		accept	
M_2	accept	reject	reject	accept	reject		accept	
M_3	reject	accept	accept	reject	accept		reject	
M_4	accept	reject	accept	accept	reject		reject	
M_5	reject	accept	reject	accept	reject		accept	
$\vdots$						$\ddots$		
D	reject	accept	reject	reject	accept		?	
$\vdots$								

ouput of $D(\langle D\rangle)$ is opposite of $M_A(\langle D, D\rangle) = D(\langle D\rangle)$

A_{TM} is undecidable

$A_{TM} = \{\langle M, w\rangle : M \text{ a TM on } \Sigma,\ M \text{ accepts } w \in \Sigma^*\}$ is undecidable

Another proof using the fact that A_{TM} is recognizable

Proof: Suppose the TM R **recognizes** A_{TM}, i.e.

$$R(\langle M, w\rangle) = \begin{cases} \textbf{Accept} & \text{if } M \text{ accepts } w \\ \textbf{Reject or Loop} & \text{if } M \text{ doesn't accept } w \end{cases}$$

Define a machine D_R that takes input encoding of a TM and works as:

$D_R(\langle M\rangle)$: Run R on $\langle M, M\rangle$ until it halts. Output the opposite answer

$$D_R(\langle M\rangle) = \begin{cases} \textbf{Reject} & \text{if } M \text{ accepts } \langle M\rangle \text{ i.e. } R(\langle M, M\rangle) \textbf{ Accepts} \\ \textbf{Accept} & \text{if } M \text{ rejects } \langle M\rangle \text{ i.e. } R(\langle M, M\rangle) \textbf{ Rejects} \\ \textbf{Loop} & \text{if } M \text{ loops on } \langle M\rangle \text{ i.e. } R(\langle M, M\rangle) \textbf{ Loops} \end{cases}$$

A_{TM} is undecidable

$A_{TM} = \{\langle M, w\rangle : M \text{ a TM on } \Sigma,\ M \text{ accepts } w \in \Sigma^*\}$ is undecidable

Another proof using the fact that A_{TM} is recognizable

Proof: Suppose the TM R **recognizes** A_{TM}, i.e.

$$R(\langle M, w\rangle) = \begin{cases} \textbf{Accept} & \text{if } M \text{ accepts } w \\ \textbf{Reject or Loop} & \text{if } M \text{ doesn't accept } w \end{cases}$$

Define a machine D_R that takes input encoding of a TM and works as:

$D_R(\langle D_R\rangle)$: Run R on $\langle D_R, D_R\rangle$ until it halts. Output the opposite answer

$$D_R(\langle D_R\rangle) = \begin{cases} \textbf{Reject} & \text{if } D_R \text{ accepts } \langle D_R\rangle \text{ i.e. } R(\langle D_R, D_R\rangle) \textbf{ Accepts} \\ \textbf{Accept} & \text{if } D_R \text{ rejects } \langle D_R\rangle \text{ i.e. } R(\langle D_R, D_R\rangle) \textbf{ Rejects} \\ \textbf{Loop} & \text{if } D_R \text{ loops on } \langle D_R\rangle \text{ i.e. } R(\langle D_R, D_R\rangle) \textbf{ Loops} \end{cases}$$

No contradiction so far, we can only tell D_H loops on $\langle D_H\rangle$

$\langle D_R, D_R\rangle \notin A_{TM}$ but $R(\langle D_R, D_R\rangle)$ loops $\qquad \triangleright$ R cannot be a decider