UNDECIDABILITY

Dr. A. Beulah
AP/CSE

RL → ReL

TM

LEARNING OBJECTIVE

- To Design Turing machines for any Languages (K3)
  - To Understand the concept of Turing Machine

REVISIT - TM

- A Turing machine (TM) is a 7-tuple

  $M = (Q \cup \{h_a, h_r\}, \Sigma, \Gamma, \delta, q_0, B, F)$ where

  – Q – A finite set of states of the finite control. Q + h_a and h_r
  – Σ – A finite set of input symbols
  – Γ – A set of tape symbols, with Σ being a subset
  – q_0 - The start state, in Q
  – B - The blank symbol in Γ, *not* in Σ (should not be an input symbol)
  – F - The set of final or accepting states

$(Q, \Sigma, \Gamma, \delta, B, q_0, F)$

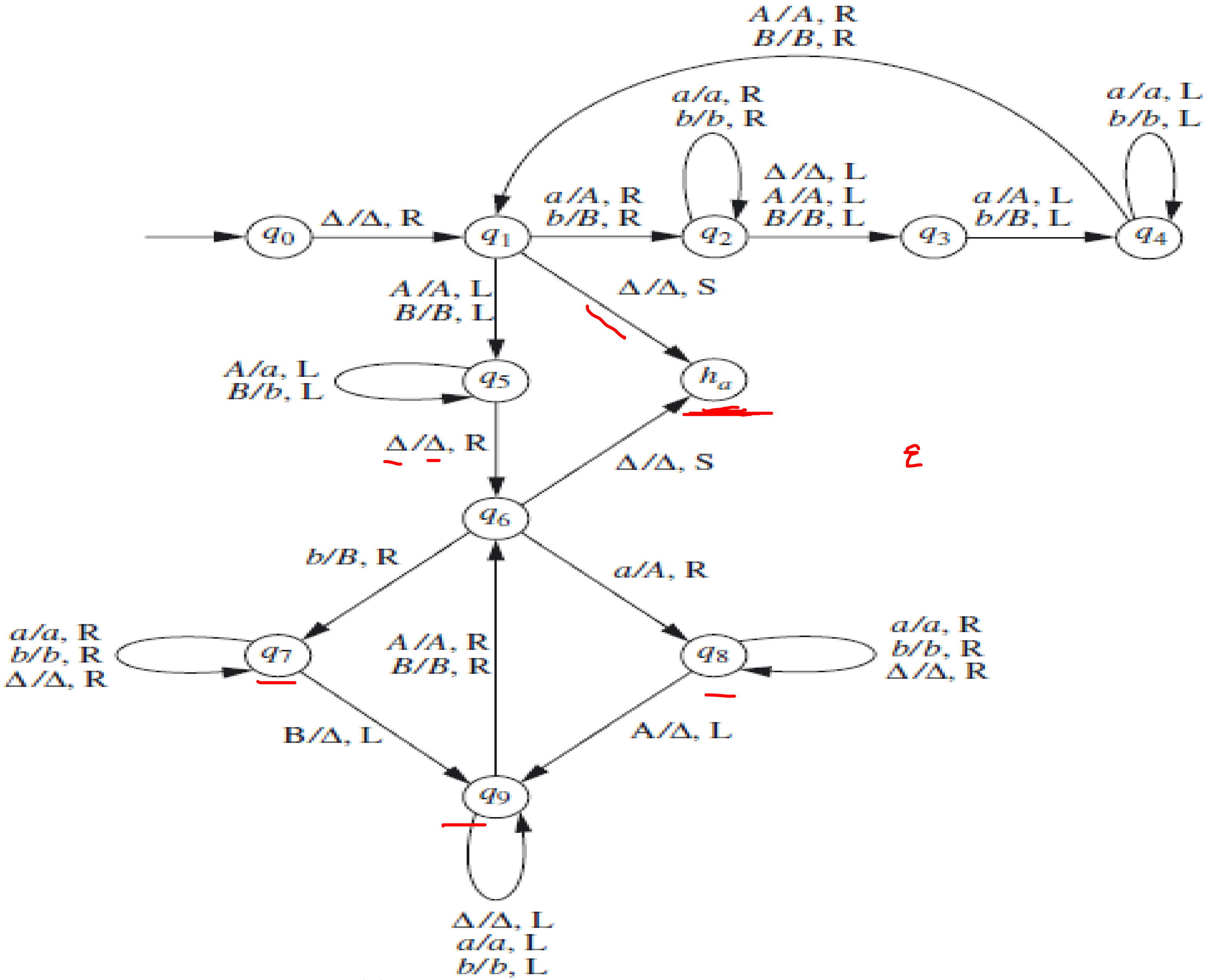
$L = \{0^n 1^n \mid n \geq 1\}$

0011

10

EXAMPLE

L = {xx |x∈ (a,b)*}



PARSING EXAMPLE WITH H_A, H_R

$$
\begin{array}{lllll}
q_0\Delta \underline{aba} & \vdash \Delta q_1 aba & \vdash \Delta A q_2 ba & \vdash^* \Delta A b a q_2 \Delta & \vdash \Delta A b q_3 a \\
 & \vdash \Delta A q_4 b A & \vdash \Delta q_4 A b A & \vdash \Delta A q_1 b A & \vdash \Delta A B q_2 A \\
 & \vdash \Delta A \underline{q_3} B A & \vdash \Delta A h_r B A & \text{(reject)} & \\
 & & & & \\
q_0\Delta \underline{ab} & \vdash \Delta q_1 ab & \vdash \Delta A q_2 b & \vdash \Delta A b q_2 \Delta & \vdash \Delta A q_3 b \Delta \\
 & \vdash \Delta q_4 A B & \vdash \Delta A q_1 B & \vdash \Delta q_5 A B & \vdash q_5 \Delta a B \\
 & \vdash \Delta q_6 a B & \vdash \Delta A \underline{q_8} B & \vdash \Delta A h_r B & \text{(reject)} \\
 & & & & \\
q_0\Delta \underline{aa} & \vdash \Delta q_1 aa & \vdash \Delta A q_2 a & \vdash \Delta A a q_2 \Delta & \vdash \Delta A q_3 a \Delta \\
 & \vdash \Delta q_4 A A & \vdash \Delta A q_1 A & \vdash \Delta q_5 A A & \vdash q_5 \Delta a A \\
 & \vdash \Delta q_6 a A & \vdash \Delta A q_8 A & \vdash \Delta q_9 A & \vdash \Delta A \underline{q_6} \Delta \\
 & \vdash \Delta A \underline{h_a} \Delta & \text{(accept)} & &
\end{array}
$$

RECURSIVE (R) AND RECURSIVELY ENUMERABLE(RE) LANGUAGES

TM < RL / ReL

DECIDABILITY VS. UNDECIDABILITY

- There are two types of TMs (based on halting):

  (*Recursive*) $L=\{0^n 1^n 2^n \mid n \geq 1\} \rightarrow TM \rightarrow$ halts $\rightarrow$ RE

  TMs that *always* halt, no matter accepting or non-accepting ≡ **DECIDABLE** PROBLEMS

  (*Recursively enumerable*) $H \rightarrow A$ M or may not halt $\rightarrow$ R   R

  TMs that *are guaranteed to halt* only on acceptance. If non-accepting, it may or may not halt (i.e., could loop forever).

  reject

- **Undecidability:**
  - Undecidable problems are those that are not recursive

ReL   Lang of const TM

RECURSIVE LANGUAGE

- A Language L is Recursive if and only if there is a TM that decides L.
  - Let $M=(Q, \Sigma, \Gamma, \delta, q_0, B, H)$ such that
    - $H= \{h_a,h_r\}$
    - $L \subseteq \Sigma^*$ Is a language
    - Assume that the initial configuration of the TM is (q_0,w)
    - M decides L if, for all strings $w \in \Sigma^*$
      - Either $w \in L$, in which case M accepts w
      - Or $w \notin L$, then M rejects w

$L \rightarrow \underset{TM}{M}$

$w \in L$ $M \rightarrow$ Accept w

$w \notin L$ $M \rightarrow$ Reject

RECURSIVE ENUMERABLE LANGUAGE

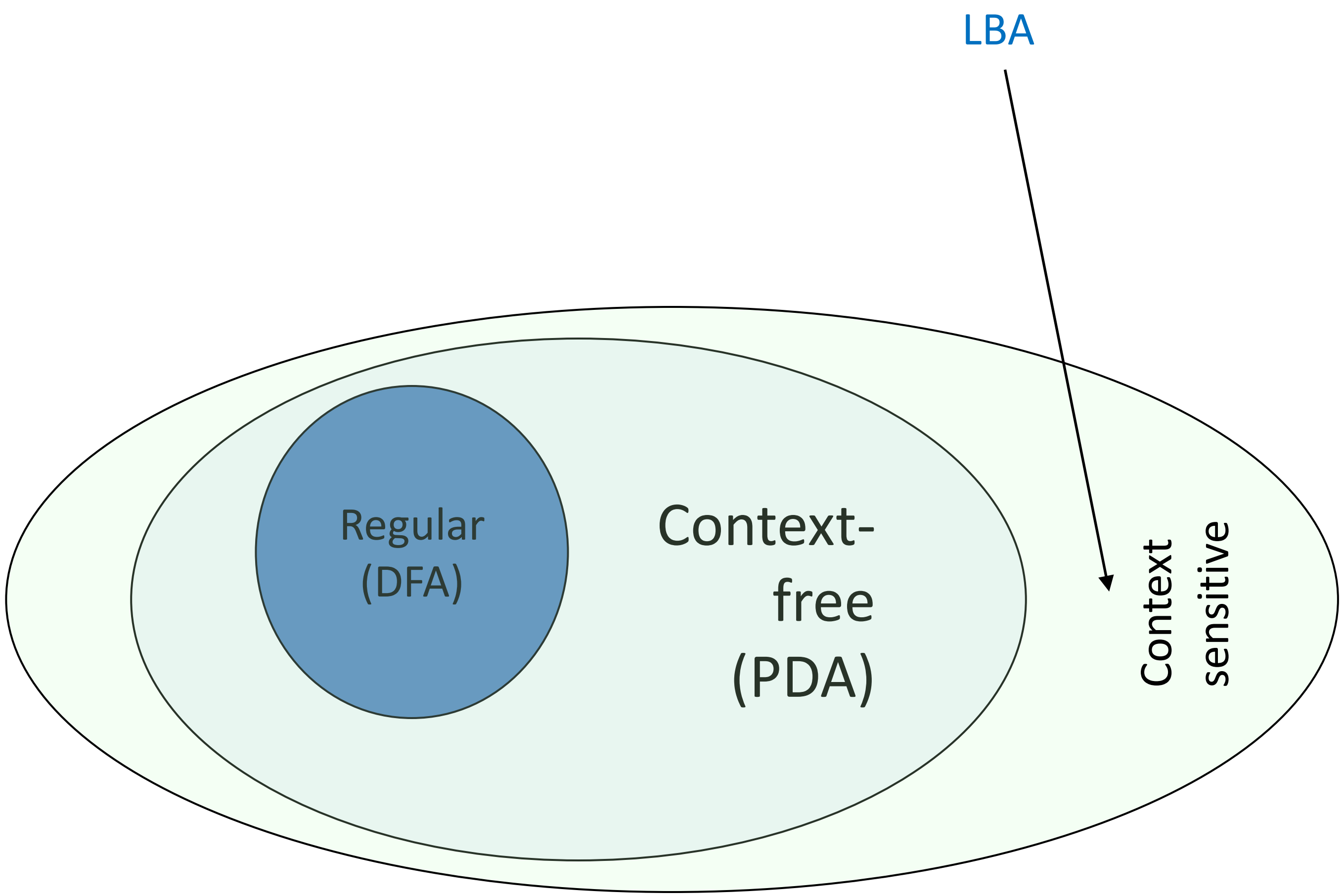
- A Language L is Recursive Enumerable if and only if there is a TM that semidecides L.
  - Let M =(Q, Σ, Γ, δ, q_0, B, H) such that
    - H= $\{h_a, h_r\}$
    - L $\subseteq \Sigma^*$ Is a language
    - Assume that the initial configuration of the TM is (q_0, w)
    - M semidecides L if, for all strings $w \in \Sigma^*$
      - Either $w \in L$, in which case M accepts w
      - Or $w \notin L$, then M does not halt

$L \to TM\_M$

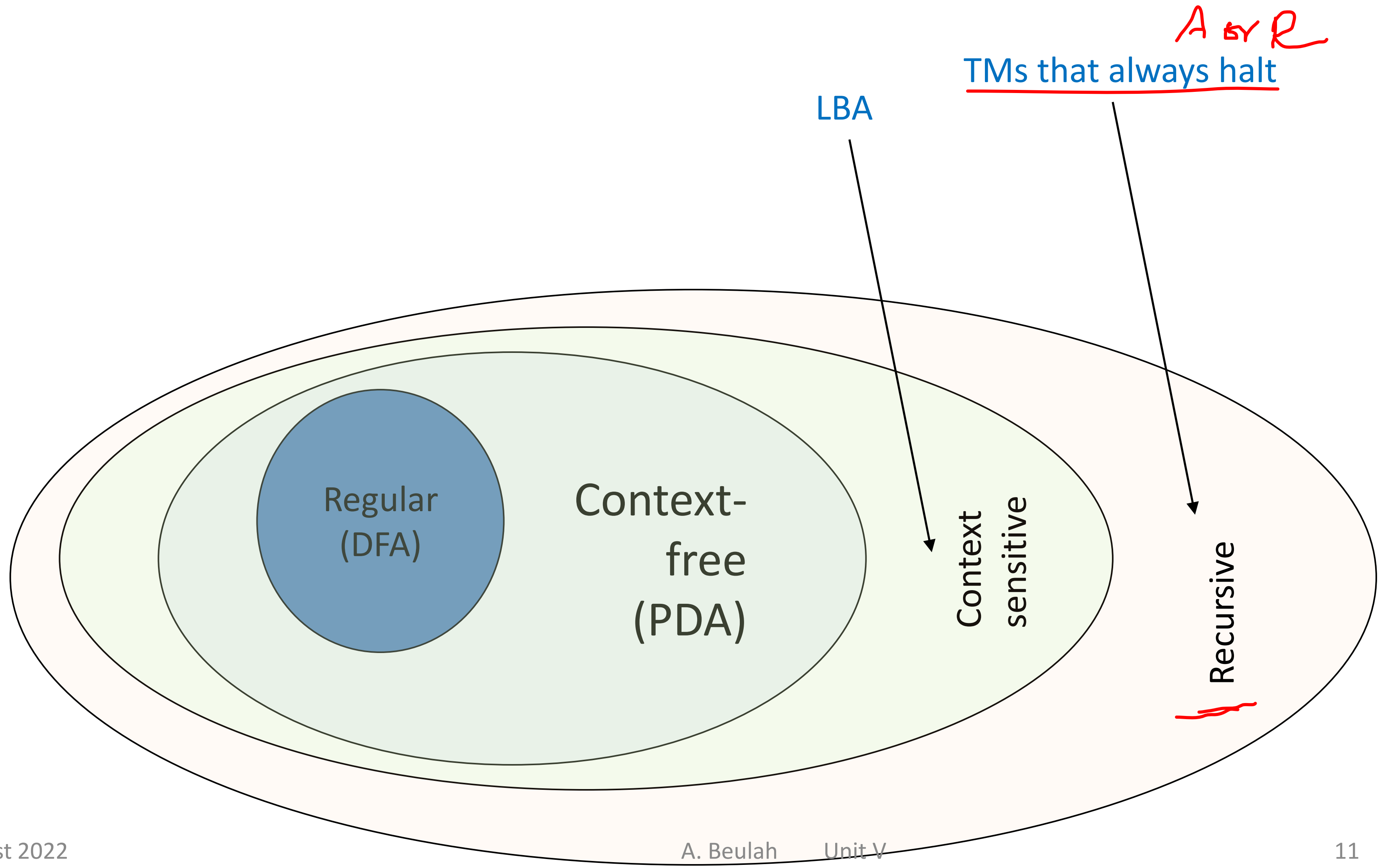
$w \in L \quad M \to \cancel{A}cce \to Halt$

$w \notin L \quad M \to \cancel{Reject}$

RECURSIVE, RE, UNDECIDABLE LANGUAGES



RECURSIVE, RE, UNDECIDABLE LANGUAGES

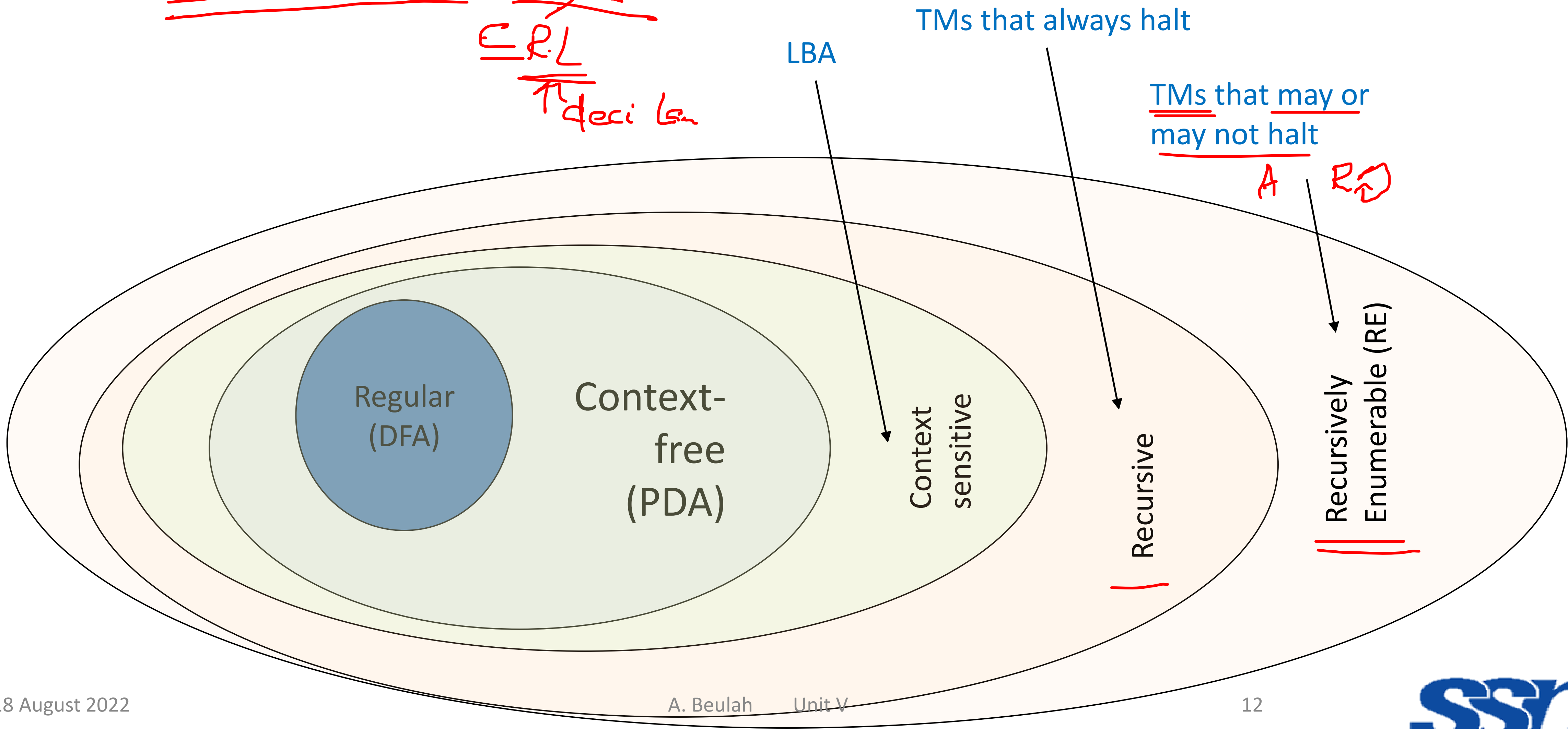


RECURSIVE, RE, UNDECIDABLE LANGUAGES

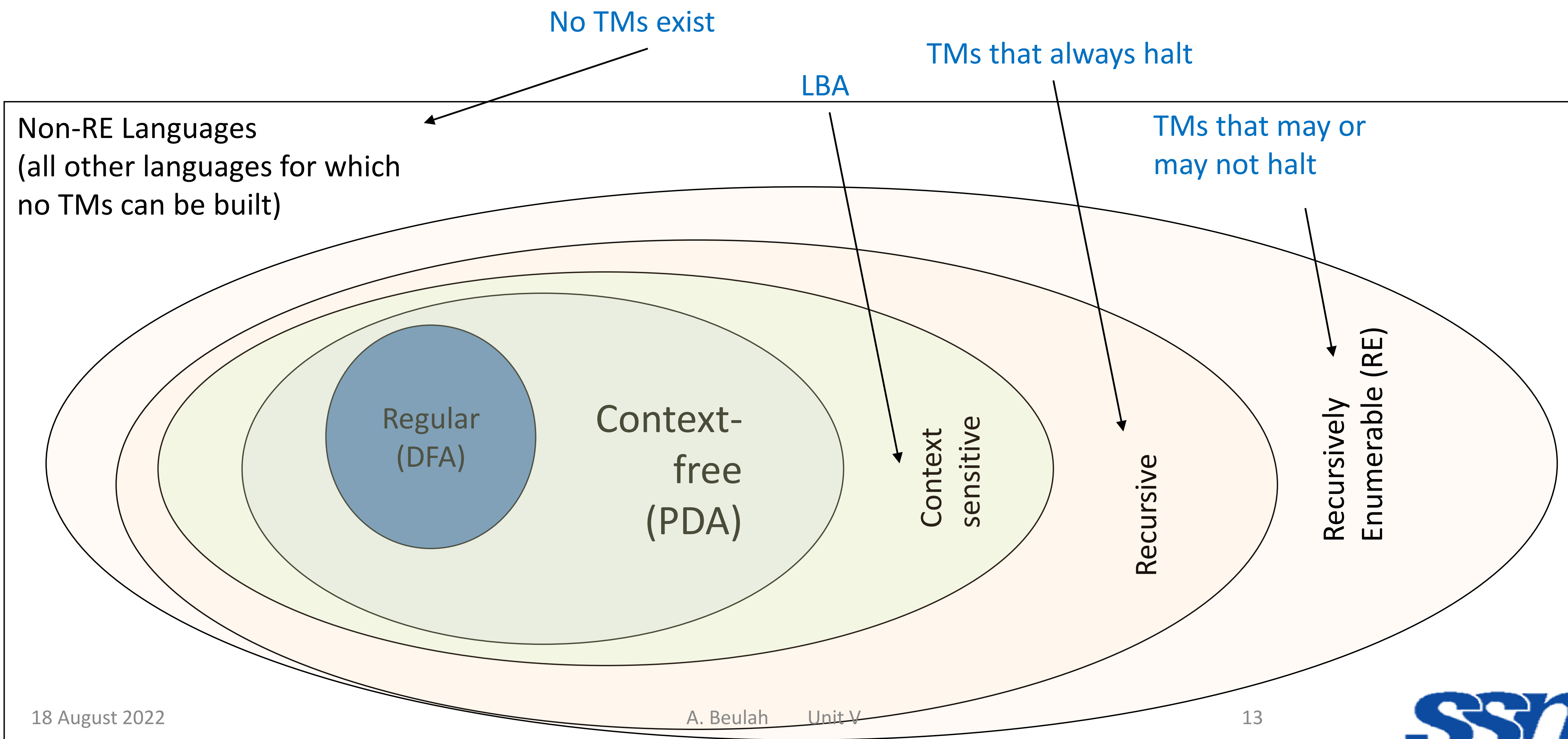
RL ⊂ CFL ⊂ CSL ⊂ ~~Re~~

⊂ R.L

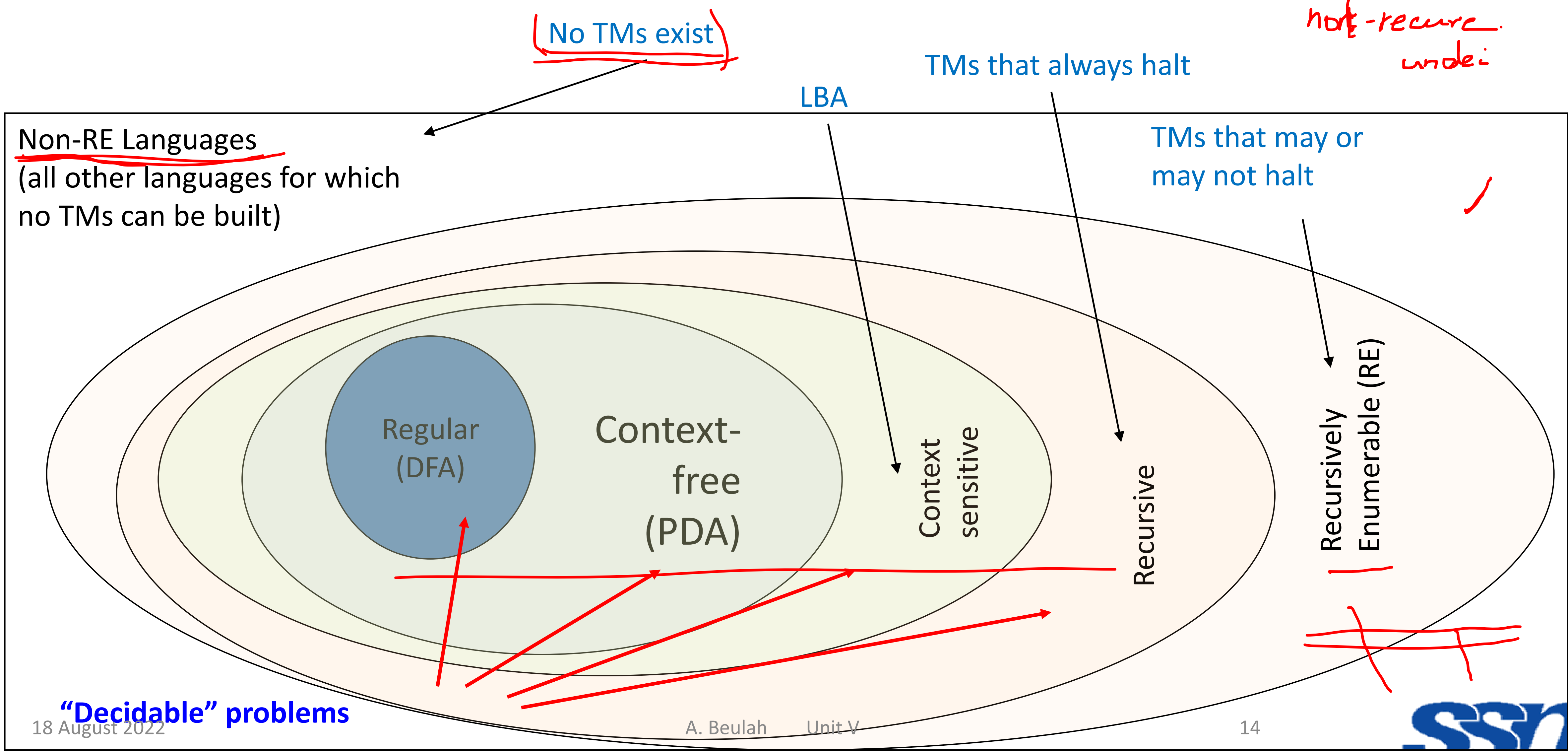
deci lan



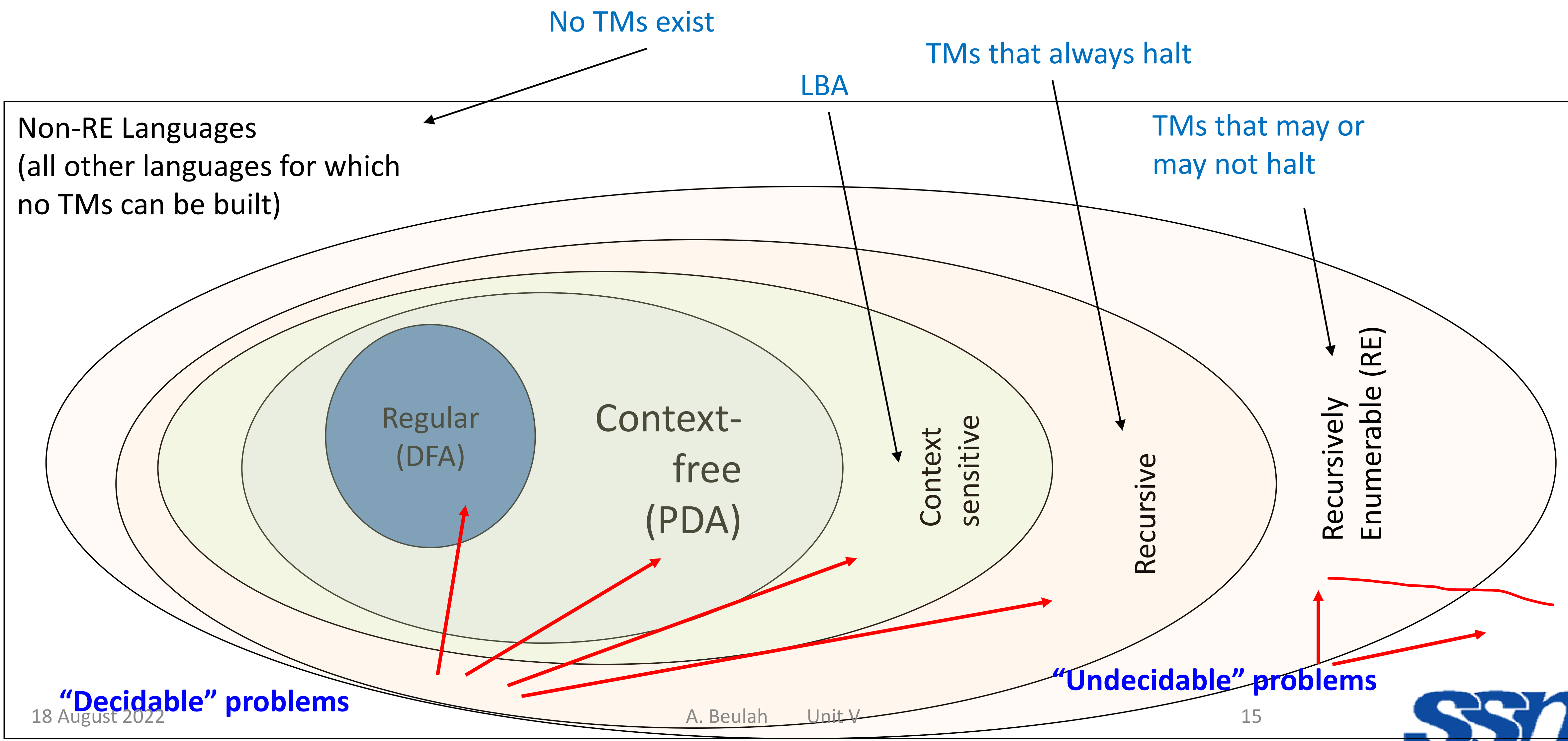
RECURSIVE, RE, UNDECIDABLE LANGUAGES



RECURSIVE, RE, UNDECIDABLE LANGUAGES



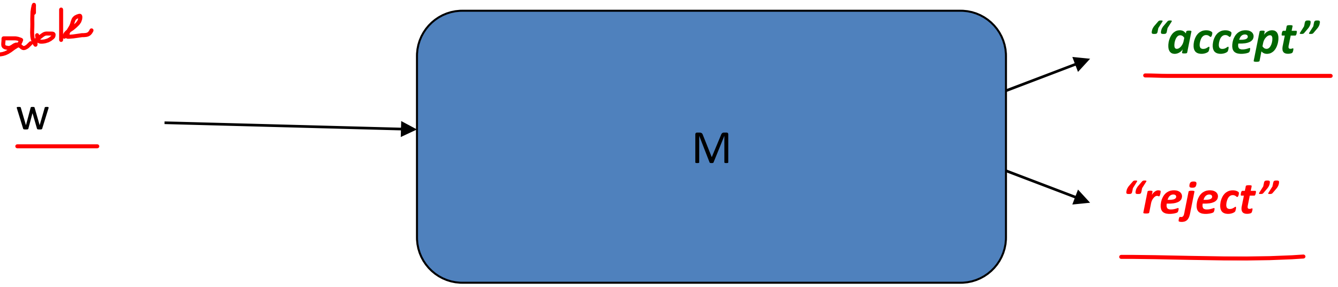
RECURSIVE, RE, UNDECIDABLE LANGUAGES



RECURSIVE, RE, UNDECIDABLE LANGUAGES

- Any TM for a <u>Recursive</u> language is going to look like this:

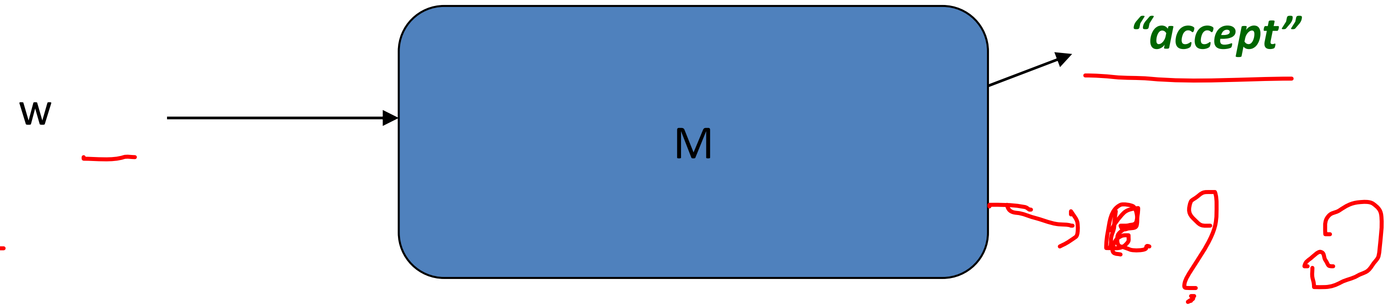
Decidable



- Any TM for a <u>Recursively Enumerable</u> (RE) language is going to look like this:

non-ReL

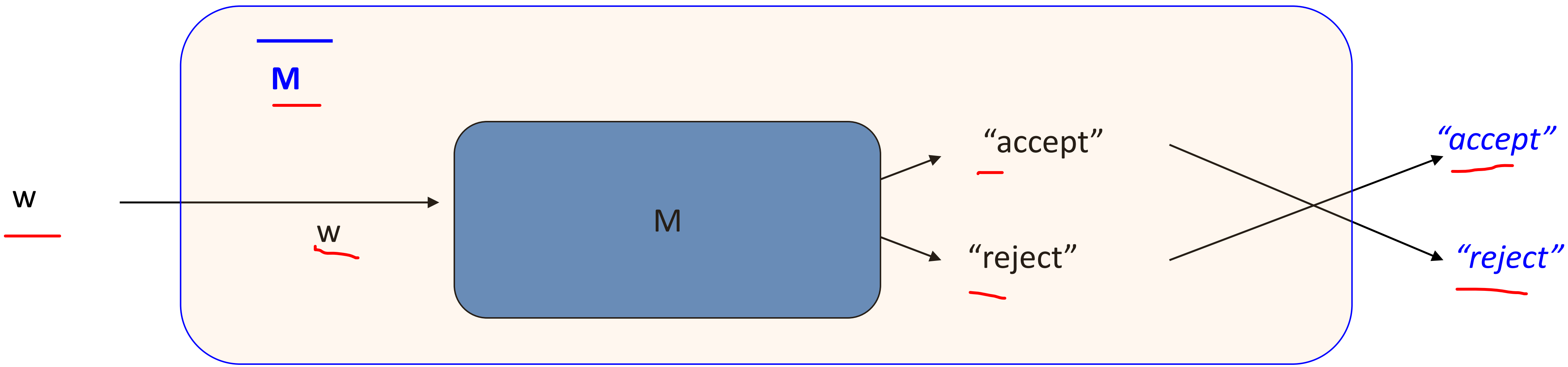
UDP TM X



CLOSURE PROPERTIES OF RECURSIVE (R) AND RECURSIVELY ENUMERABLE(RE) LANGUAGES

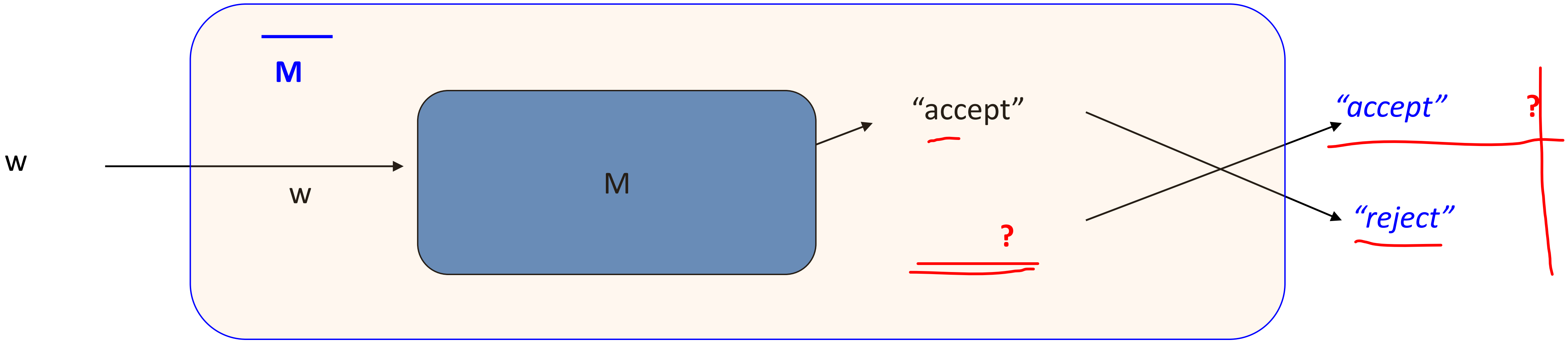
RL ARE CLOSED UNDER COMPLEMENTATION

– If L is Recursive, $\overline{L}$ is also Recursive



ARE ~~RL~~ REL CLOSED UNDER COMPLEMENTATION? (NO)

– If L is RE, $\overline{L}$ need not be RE

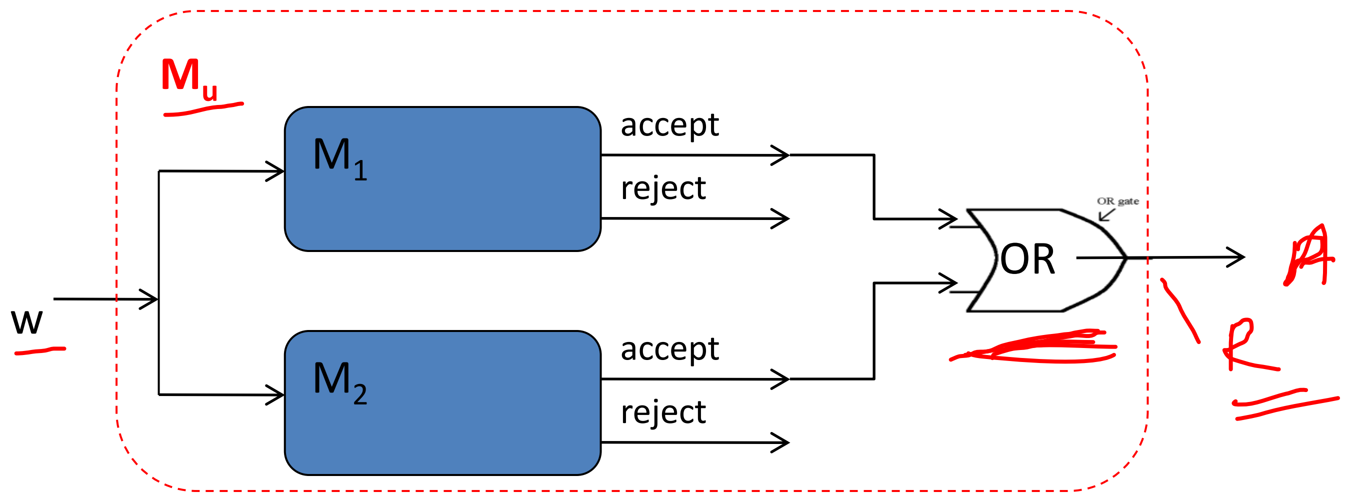


RECURSIVE LANGS ARE CLOSED UNDER UNION

$L_1 \cup L_2$

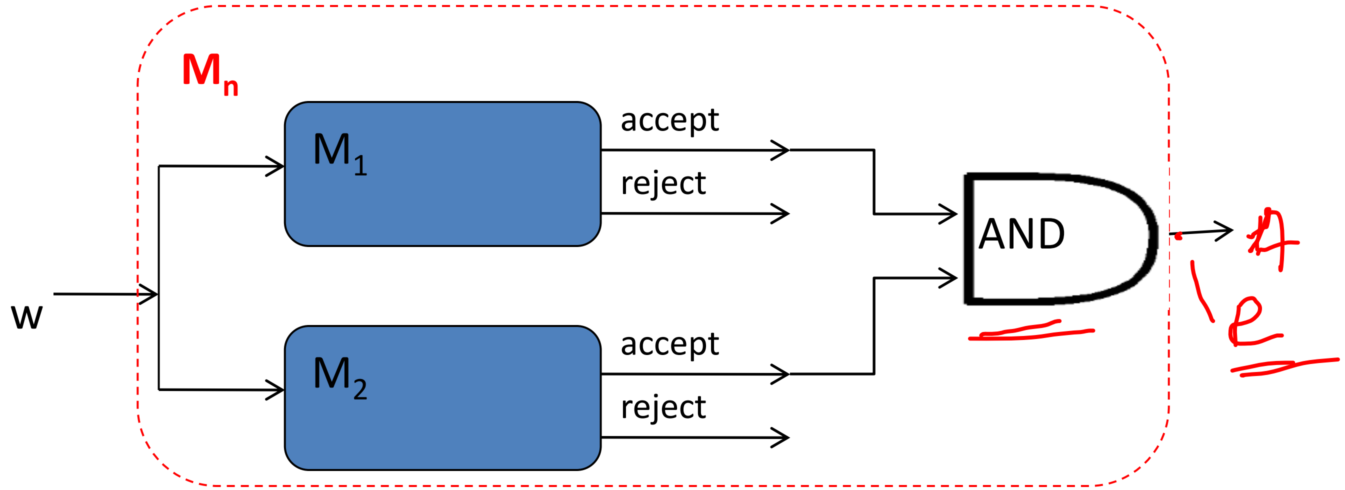
M_1 M_2

- Let M_u = TM for L_1 U L_2
- M_u construction:
  1. Make 2-tapes and copy input w on both tapes
  2. Simulate M_1 on tape 1
  3. Simulate M_2 on tape 2
  4. If either M_1 or M_2 accepts, then M_u accepts
  5. Otherwise, M_u rejects.



RL ARE CLOSED UNDER INTERSECTION

- Let M_n = TM for $L_1 \cap L_2$
- M_n construction:
  1. Make 2-tapes and copy input w on both tapes
  2. Simulate M_1 on tape 1
  3. Simulate M_2 on tape 2
  4. If either M_1 AND M_2 accepts, then M_n accepts
  5. Otherwise, M_n rejects.



OTHER CLOSURE PROPERTY RESULTS

- Recursive languages are also closed under:
  - Concatenation
  - Kleene closure (star operator)
  - Homomorphism, and inverse homomorphism
- RE languages are closed under:
  - Union, intersection, concatenation, Kleene closure

- RE languages are *not* closed under:
  - complementation

TEST YOUR KNOWLEDGE

- Let L1 be a recursive language. Let L2 and L3 be languages that are recursively enumerable but not recursive. Which of the following statements is not necessarily true?
  - L2 – L1 is recursively enumerable.
  - L1 – L3 is recursively enumerable
  - L2 ∩ L1 is recursively enumerable
  - L2 ∪ L1 is recursively enumerable

SUMMARY

- What is undecidability
- Recursive and Recursive enumerable languages

REFERENCE

- Hopcroft J.E., Motwani R. and Ullman J.D, “Introduction to Automata Theory, Languages and Computations”, Second Edition, Pearson Education, 2008